Advances in Intelligent Systems and Computing 552

Ajith Abraham · Abdelkrim Haqiq
Adel M. Alimi · Ghita Mezzour
Nizar Rokbani · Azah Kamilah Muda
Editors

Proceedings of the 16th International Conference on Hybrid Intelligent Systems (HIS 2016)

Springer

Advances in Intelligent Systems and Computing

Volume 552

Series editor

Janusz Kacprzyk, Polish Academy of Sciences, Warsaw, Poland
e-mail: kacprzyk@ibspan.waw.pl

About this Series

The series "Advances in Intelligent Systems and Computing" contains publications on theory, applications, and design methods of Intelligent Systems and Intelligent Computing. Virtually all disciplines such as engineering, natural sciences, computer and information science, ICT, economics, business, e-commerce, environment, healthcare, life science are covered. The list of topics spans all the areas of modern intelligent systems and computing.

The publications within "Advances in Intelligent Systems and Computing" are primarily textbooks and proceedings of important conferences, symposia and congresses. They cover significant recent developments in the field, both of a foundational and applicable character. An important characteristic feature of the series is the short publication time and world-wide distribution. This permits a rapid and broad dissemination of research results.

Advisory Board

More information about this series at http://www.springer.com/series/11156

Ajith Abraham · Abdelkrim Haqiq
Adel M. Alimi · Ghita Mezzour
Nizar Rokbani · Azah Kamilah Muda
Editors

Proceedings of the 16th International Conference on Hybrid Intelligent Systems (HIS 2016)

Springer

Editors
Ajith Abraham
(MIR Labs)
Machine Intelligence Research Labs
Auburn, WA
USA

Abdelkrim Haqiq
Hassan 1st University
Settat
Morocco

Adel M. Alimi
ENIS
University of Sfax
Sfax
Tunisia

Ghita Mezzour
Technopolis Rabat-Shore Rocade
International University of Rabat
Sala el Jadida
Morocco

Nizar Rokbani
Inst Applied Dept of Electronics, Taffala
University of Sousse
Sousse
Tunisia

Azah Kamilah Muda
Fakulti Teknologi Maklumat dan Komunikas
Universiti Teknikal Malaysia Melaka
Durian Tunggal
Malaysia

ISSN 2194-5357 ISSN 2194-5365 (electronic)
Advances in Intelligent Systems and Computing
ISBN 978-3-319-52940-0 ISBN 978-3-319-52941-7 (eBook)
DOI 10.1007/978-3-319-52941-7

Library of Congress Control Number: 2017932620

Printed on acid-free paper

This Springer imprint is published by Springer Nature
The registered company is Springer International Publishing AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

HIS 2016 - Welcome Message

Welcome to Marrakech, Morocco and to the 16th International Conference on Hybrid Intelligent Systems (HIS 2016) and the 8th World Congress on Nature and Biologically Inspired Computing (NaBIC 2016), held during November 21–23, 2016. HIS - NaBIC 2016 is jointly organized by the Machine Intelligence Research Labs (MIR Labs), USA; Hassan 1st University, Settat, Morocco and University of Sfax, Tunisia.

Hybridization of intelligent systems is a promising research field of modern artificial/computational intelligence concerned with the development of the next generation of intelligent systems. A fundamental stimulus to the investigations of Hybrid Intelligent Systems (HISs) is the awareness in the academic communities that combined approaches will be necessary if the remaining tough problems in computational intelligence are to be solved. Recently, hybrid intelligent systems are getting popular due to their capabilities in handling several real-world complexities involving imprecision, uncertainty, and vagueness. HIS 2016 builds on the success of HIS 2015, which was held in Seoul, Korea during November 16–18, 2016. Nature and Bio-inspired Computing is currently one of the most exciting research areas, and it is continuously demonstrating exceptional strength in solving complex real-life problems. The main driving force of the conference is to further explore the intriguing potential of Bio-inspired Computing. NaBIC 2016 builds on the success of NaBIC 2015, which was held in Pietermaritzburg, South Africa during December 01–03, 2015.

Many people have collaborated and worked hard to produce this years successful HIS - NaBIC conferences. First, we would like to thank all the authors for submitting their papers to the conference, for their presentations and discussions during the conference. Our thanks to Program Committee members and reviewers, who carried out the most difficult work by carefully evaluating the submitted papers. We have two plenary speakers:

- Oscar Castillo, Tijuana Institute of Technology, Tijuana, Mexico
- Layth Sliman, EFREI, France

Thanks to all the speakers for their valuable time. The themes of the contributions and scientific sessions range from theories to applications, reflecting a wide spectrum of coverage of the hybrid intelligent systems and computational intelligence areas. HIS - NaBIC 2016 received 100+ submissions from over 25 countries, and each paper was reviewed by five or more reviewers in a standard peer-review process. Based on the recommendation by five independent referees, finally 58 papers were accepted for publication in the proceedings.

We would like to thank the Springer Publication team for the wonderful support for the publication of this volume.

HIS – NaBIC 2016 General Chairs
Abdelkrim Haqiq, Hassan 1st University, Settat, Morocco
Ajith Abraham, MIR Labs, USA
Adel M. Alimi, University of Sfax, Tunisia
HIS – NaBIC 2016 Program Committee Chairs
Ghita Mezzour, International University of Rabat, Morocco
Nizar Rokbani, University of Sousse, Tunisia
Amine Berqia, ENSIAS, Mohammed V University, Rabat, Morocco

Organization

Honorary Chairs

Mohamed Knidiri	President of Association le Grand Atlas, President of the Private University of Marrakesh, Morocco
Ahmed Nejmeddine	President of Hassan 1st University, Settat, Morocco
Houssine Bouayad	Dean of FST, Hassan 1st University, Settat, Morocco

General Chairs

Abdelkrim Haqiq	Hassan 1st University, Settat, Morocco
Ajith Abraham	MIR Labs, USA
Adel M. Alimi	University of Sfax, Tunisia

Program Committee Chairs

Ghita Mezzour	International University of Rabat, Morocco
Nizar Rokbani	University of Sousse, Tunisia

Organizing Chairs

Mohamed Chakraoui	Multidisciplinary Faculty of Ouarzazate, Morocco
Jaouad Dabounou	FST, Hassan 1st University, Settat, Morocco

Organizing Committee

Ali Alouai	FST, Hassan 1st University, Settat, Morocco
Ayoub Barakat	FST, Hassan 1st University, Settat, Morocco
Abdessamad Bouqdir	FST, Hassan 1st University, Settat, Morocco
Yassir El Filali	FST, Hassan 1st University, Settat, Morocco
Youmna El Hiss	FST, Hassan 1st University, Settat, Morocco
Ayman Hadri	FST, Hassan 1st University, Settat, Morocco
Amine Maarouf	FST, Hassan 1st University, Settat, Morocco

Publication Chair

Azah Kamilah Muda Universiti Teknikal Malaysia Melaka, Malaysia

Advisory Board members

Aboul Ella Hassanien	University of Cairo, Egypt
André Ponce de Leon F de Carvalho	University of Sao Paulo, Brazil
Andries Engelbrecht	University of Pretoria, South Africa
Emilio Corchado	Universidad de Salamanca, Spain
Francisco Herrera	University of Granada, Spain
Francesco Marcelloni	University of Pisa, Italy
Hideyuki Takagi	Kyushu University, Japan
Imre J. Rudas	Obuda University, Hungary
Jeng-Shyang Pan	Harbin Institute of Technology, China
Krzysztof Cios	Virginia Commonwealth University, USA
Mario Koeppen	Kyushu Institute of Technology, Japan
Janusz Kacprzyk	Polish Academy of Sciences, Poland
Sebastián Ventura	University of Cordoba, Spain

Web Administrator

Kun Ma University of Jinan, China

Technical Program Committee

Abdellah Najid	Institut National des Postes et Télécommunications, Rabat, Morocco

Abdellah Touhafi	Vrije Universiteit Brussel, Belgium
Ahlame Begdouri	My Abdellah University, Fès, Morocco
Alberto Cano	Virginia Commonwealth University, Virginia
Alex James	Nazarbayev University, Kazakhstan
Alexander Mendiburu	The University of the Basque Country, Spain
Alfredo Arias Montano	Instituto Politécnico Nacional, Mexico
Amelia Zafra Gómez	University of Córdoba, Spain
Amine Berqia	Mohammed V Souissi University, Rabat, Morocco
Amit Dutta	Barkatullah University, India
Anan Banharnsakun	King Mongkut's University of Technology Thonburi, Thailand
Anand Jayant Kulkarni	Nanyang Technological University, Singapore
Anas Abouelkalam	Ecole Nationale des Sciences Appliquées, Morocco
Andrea Corradini	University of Southern Denmark, Denmark
Andrea Schaerf	University of Udine, Italy
Andrew L. Nelson	Androtics LLC, USA
Anna Kononova	Heriot-Watt University, UK
Annabel Latham	Manchester Metropolitan University, UK
Anne Laurent	University of Montpellier II, France
Antonio Dourado	University of Coimbra, Portugal
Antonio J. Tallón Ballesteros	Universidad de Sevilla, Spain
Antonio LaTorre	Universidad Politécnica de Madrid, Spain
Antreas Nearchou	University of Patras, Greece
Arshin Rezazadeh	Shahid Chamran University of Ahvaz, Iran
Azah Kamilah Muda	Universiti Teknikal Malaysia Melaka, Malaysia
Azeddine Bilami	Batna University, Algeria
Bahareh Asadi	Islamic Azad University, Iran
Baihai Zhang	Beijing Institute of Technology, China
Biju Issac	Teesside University, UK
Bing Xue	Victoria University of Wellington, New Zealand
Binod Kumar Prasad	Maharastra Academy of Engineering, India
Carlos Alberto Ochoa Ortiz	Universidad Autónoma de Ciudad Juárez, Mexico
Carlos Fernandez Llatas	Universidad Politecnica de Valencia, Spain
Carlos Pereira	Instituto Superior de Engenharia de Coimbra, Portugal
Cerasela Crisan	"Vasile Alecsandri" University of Bacau, Romania
Cesar Torres	Centro de Investigación y de Estudios Avanzados del Instituto Politécnico Nacional, Mexico
Cherki Daoui	Faculty of Science and Technology, BP 523, Beni Mellal, Morocco
Chi Kin Chow	University of Hong Kong, China

Chu Hsing Lin	Tunghai University, Taiwan
Chun Wei Lin	National University of Kaohsiung, Taiwan
Crina Grosan	Babeş-Bolyai University, Romania
Daniela Zaharie	West University of Timisoara, Romania
Diaf Moussa	Université Mouloud Mammeri, Algeria
Driss Bouzidi	Mohammed V University, Morocco
Driss El Ouadghiri	Moulay Ismail University, Meknès, Morocco
Eduardo Solteiro Pires	University of Trás-os-Montes and Alto Douro, Portugal
Eiji Uchino	Yamaguchi University, Japan
Elpida Tzafestas	University of Athens, Greece
El-Sayed M. El-Alfy	King Fahd University of Petroleum and Minerals, Saudi Arabia
Ender Ozcan	University of Nottingham, UK
Essaïd Sabir	Hassan II University of Casablanca, Morocco
Fernando J. Von Zuben	Universidade Estadual de Campinas, Brazil
Fernando Jimenez	University of Murcia, Spain
Firkhan Ali Hamid Ali	Universiti Tun Hussein Onn Malaysia, Malaysia
Francesco Moscato	Seconda Università degli Studi di Napoli, Italy
Francisco Chicano	Universidad de Málaga, Spain
Franco Frattolillo	Università degli Studi del Sannio, Italy
Gabriel Luque	Universidad de Málaga, Spain
Gai Ge Wang	Jiangsu Normal University, China
Gazi Erkan Bostanci	University of Ankara, Turkey
Georg Peters	Munich University of Applied Sciences, Germany
George Georgiev	University of Wisconsin Oshkosh, USA
Ghanshyam Thakur	National Institute of Technology, India
Ghita Mezzour	Carnegie Mellon University, USA
Ghizlane Orhanou	Mohammed V University, Rabat, Morocco
Giancarlo Mauri	Università di Milano Bicocca, Italy
Hajar Mousannif	Université Cadi Ayyad, Morocco
Heder Bernardino	Universidade Federal de Juiz de For a, Brazil
Hongwei Mo	Harbin Engineering University, China
Ibibia.K. Dabipi	University of Maryland Eastern Shore, Maryland
Irene Moser	Swinburne University of Technology, Australia
Jaouad Dabounou	Faculte des Sciences et Techniques de Settat, Morocco
José Raúl Romero	University of Córdoba, Spain
Jose Santos	Universidad de A Coruña, Spain
Karmela Aleksic – Maslac	Zagreb School of Economics and Management, Zagreb, Croatia
Keun Ho Ryu	Chungbuk National University, South Korea

Khalid Zine-Dine	Chouaib Doukali University, El Jadida, Morocco
Konstantinos Parsopoulos	University of Ioannina, Greece
Korhan Karabulut	Yaşar Üniversitesi, Turkey
Kun Ma	University of Jinan, China
Laurence Amaral	Universidade Federal de Uberlândia, Brazil
Li Zhang	Soochow University, China
M. Hefnawi	Royal Military College of Canada, Canada
Man Leung Wong	Lingnan University, China
Mario Koeppen	Kyushu Institute of Technology, Japan
Marjan Kuchaki Rafsanjani	Shahid Bahonar University of Kerman, Iran
Mohamed Bakhouya	International University of Rabat, Morocco
Mohamed Hanini	Hassan 1st University, Settat, Morocco
Mohamed Nemiche	Ibn Zohr University, Morocco
Mohamed Sabbane	Moulay Ismail University, Morocco
Mohamed El kamili	My Abdellah University, Fès, Morocco
Mohamed Moughit	Hassan 1st University, Settat, Morocco
Mohammed Ouzzif	Hassan II University of Casablanca, Morocco
Mohammed Erradi	Mohammed V University, Rabat, Morocco
Mostafa Belkasmi	Mohammed V University, Rabat, Morocco
Mostapha Zbakg	Mohammed V University, Rabat, Morocco
Nabil Laachfoubi	Hassan 1st University, Settat, Morocco
Nassereddine Bouchaib	Hassan 1st University, Settat, Morocco
Ounsa Roudies	Mohammed V University, Rabat, Morocco
Patrick Siarry	Universit de Paris, France
Rachid El-Azouzi	Université d'Avignon et des Pays de Vaucluse, France
Rachida Ajhoun	Mohammed V University, Rabat, Morocco
Ricardo Landa	Instituto Politécnico Nacional, Mexico
Richard Duro	Universidad de A Coruña, Spain
Roberto Antonio Vázquez Espinoza De Los Monteros	Lasallistas de Corazón, Mexico
Rodica Ioana Lung	Babeş-Bolyai University, Romania
Said El Kafhali	Hassan 1st University, Settat, Morocco
Shyam Sundar	National Institute of Technology Raipur, India
Slimane Mekaoui	Telecommunications Department, USTH, Algiers, Algeria
Sofia Douda	Faculté des Sciences et Techniques, Settat, Morocco
Soumia Ichoua	Embry-Riddle Aeronautical University, Florida
Sung Bae Cho	Yonsei University, South Korea
Susana C. Esquivel	Universidad Nacional de San Luis, Argentina
Suwin Sleesongsom	Chiangrai College, Thailand
Tatiana Kalganova	Brunel University London, UK

Thomas Hanne	University of Applied Sciences Northwestern Switzerland, Switzerland
Tien Tsin Wong	University of Hong Kong, China
Tzung Pei Hong	National University of Kaohsiung, Taiwan
Victor Manuel Landassuri Moreno	Centro Universitario UAEM Valle de México, Mexico
Vincenzo Piuri	Università degli Studi di Milano, Italy
Wei Chiang Hong	Oriental Institute of Technology, Taiwan
Yi Mei	RMIT University, Australia
Ying Ping Chen	National Chiao Tung University, Taiwan
Yun Huoy Choo	Universiti Teknikal Malaysia Melaka, Malaysia

Organizers

Sponsors

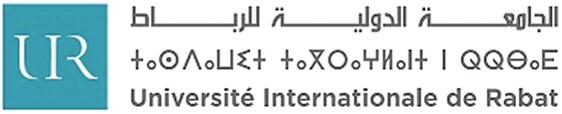

جامعة الحسن الأول
UNIVERSITÉ HASSAN 1er


كلية العلوم
و التقنيات سطات
FACULTÉ DES SCIENCES
ET TECHNIQUES SETTAT


e-IGN
.ORG


الأطلس
الكبير
Association
Le Grand Atlas


GFC
Groupement de Formation Continue


UR
الجامعة الدولية للرباط
ⵜⴰⵙⴷⴰⵡⵉⵜ ⵜⴰⵅⵔⵕⵕⵉⵜ ⵏ ⵕⴱⴰⵟ
Université Internationale de Rabat

Contents

Hand Gesture Recognition Using Color Markers

Mubashira Zaman, Soweba Rahman, Tooba Rafique, Filza Ali, and Muhammad Usman Akram(✉)

Department of Computer Engineering, College of Electrical and Mechanical Engineering, National University of Sciences and Technology, Islamabad, Pakistan
mobizmn@live.com, sr962014@gmail.com, rafiquetooba@yahoo.com, filza.0023@hotmail.com, usmakram@gmail.com

Abstract. In the last few years, many methods have been proposed for gesture recognition but even now, there is not a standard method set for this type of detection. Both online and offline approaches can be used for this purpose. This paper uses the offline method with the aim to detect the 26 letters of the American Sign Language (ASL). First of all, a glove with 3 color markers, namely red, green and yellow is worn upon the hand. Then the training and testing sets are acquired. Then, masking of the three colors is done followed by noise removal. The next step is to find centroids of the white regions in each mask. Afterwards, distances between centroids are calculated and then an algorithm is developed based on the number of regions in red, green and yellow masks and the distances between centroids to detect the letters. This method has an overall accuracy of 96%.

Keywords: Color pointers · Hand gesture recognition · American sign language · Glove

1 Introduction

According to an article written in 1983, the total number and deaf-mutes (deaf and dumb) at that time was found to be ranging from 0.7 million to 0.9 million and out of these people, 63% were born with this disability. [1] While currently, according to the World Federation of the Deaf, there are almost 70 million deaf people in the world [2].

The origin of the first sign language is said to be accredited to Abbe Charles Michel de L'epee who invented French sign language and taught it in his first established teaching school for the deaf. Later on, the American Sign Language was invented by Thomas Hopkins in the early 1800's who set up an American School for the deaf in 1917 [3].

Sign language is the major means of communication for deaf people. When communicating with such people, we often find ourselves lost in trying to understand what they are trying to say because most people don't understand the sign language. So, gesture recognition for sign language is such a tool that if devised correctly, can be

A. Abraham et al. (eds.), *Proceedings of the 16th International Conference on Hybrid Intelligent Systems (HIS 2016)*, Advances in Intelligent Systems and Computing 552,
DOI 10.1007/978-3-319-52941-7_1

used to help many people. The technique can further be incorporated to recognize numbers and another gestures which can then be used in artificial intelligence (AI).

The proposed system will consist of laptop interface, the user will wear a black glove with different colored tapes (red, green, yellow) on each finger. Then, hand gestures of sign language are made and captured with camera (Fig. 1).

Fig. 1. Hand gestures of A, B, C

The organization of report is as follows. Section 2 describes the related work done in the field of gesture recognition. Section 3 explain the proposed technique. Section 4, tells about the acquired results and finally, Sect. 5 concludes the report.

2 Related Work

A lot of work has been done before on the topic of gesture recognition so below are explained some of the approaches that were taken towards this particular problem.

One approach was to use a data glove with 10 tilt sensors to detect movement of finger muscles, an accelerometer to detect overall movement of hand, and a micro-controller and a Bluetooth device to send the data to a mobile phone. This method had a variable background and an online approach. A total of 120 attempts were taken which generated 89% accuracy [4].

Another method used was detection through Karhunen-Loeve Transform. Firstly, the skin was filtered. Then the palms of hands were cropped. It was followed by edge detection, feature extraction and finally, the classification was done. The method had a static background and an offline approach. A total of 100 test images generated an accuracy of 96%. [5] Another approach using SIFT algorithm, having a variable background and working for both online and offline approaches used 80% data for training and 20% for testing. The method produced 100% accuracy [6].

Dong et al. proposed a method of alphabet recognition using Microsoft Kinect. Static images were taken for training. A latex color glove with 11 color markers was used. An accuracy of 90% was achieved on the Surrey University's dataset. [7] Another research used the same method for robust hand gesture detection. It had a static background and an offline approach. The data set of 1000 images yielded 93% accuracy [8].

Bag-of-features and multiclass support vector machine (SVM) was also used in an approach. It also used SIFT and K-means clustering. An SVM classifier was used for training. It used a variable background and an online approach with a training set of 600 images and an accuracy of 96%. [9] Another technique used was combining RGB and time-of-flight (ToF) cameras and devising an algorithm based on depth and color. It used a variable background and an online approach. A total data of 350 images for 6 gestures was taken such that have was for training and half for testing. It generated 100% accurate results for gesture recognition [10].

3 Proposed System

In the proposed hand gesture recognition system we first capture three samples of each alphabet. Figure 3 shows the images taken for each letter. After resizing these images, we convert each image into HSV. After that, we find out separate masks for each of the colors; red, yellow, green. The next step is to remove noise from the masks. Once the masked images are obtained, centroids of each region in a mask are calculated and the distance between all centroids are found. The look-up table is formed and at the end, we compare the data of the test image with lookup table. Figure 2 shows the flow chart of proposed system.

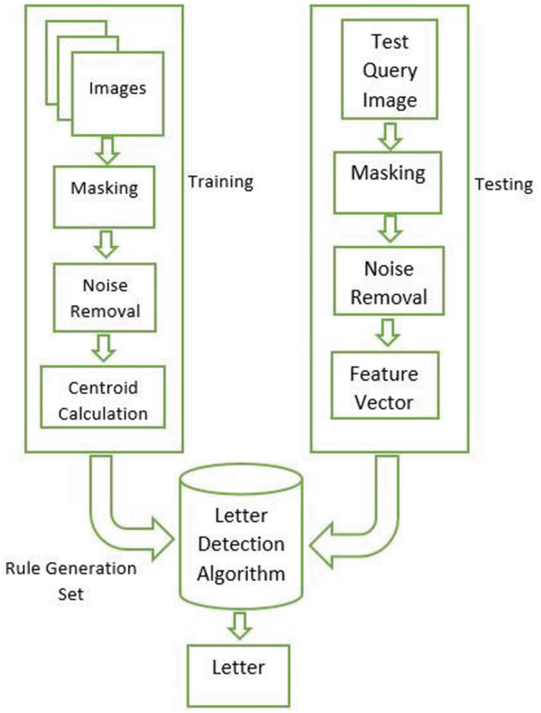


Fig. 2. Proposed system for sign language detection

Fig. 3. American sign language

3.1 Image Acquisition

First a black glove was worn and color tapes (red, green and yellow) are applied on it. Then images of sign language of alphabets from A–Z were taken with a camera. Three to five samples of each alphabet were taken. After reading the images, they are resized to smaller dimensions. Figure 4 shows the three samples taken for the letter L.

Fig. 4. Training images for L

3.2 HSV and Masking

For one image, its HSV image is obtained which is used to calculate masks for red, green and yellow color. In each mask, thresholding is applied so that the main color is equated to 1 and the rest to 0 (Fig. 5).

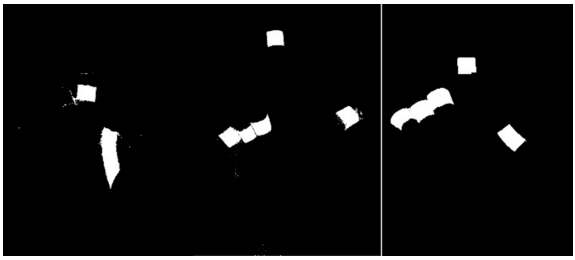


Fig. 5. (a) Red Mask (b) Green Mask (c) Yellow Mask

3.3 Noise Removal

When masks are obtained they contain a certain level of noise which is removed by using erosion and morphological techniques. Kernels of any size are chosen for each of the following techniques in relevance with the noise in the images (Fig. 6).

- Erosion: It removes layers of pixels from the inner and outer boundary of the regions; thus removing any white noise present in the image.
- Morphological opening: It first erodes the image by removing inside and outside layers and then dilates it by adding an inside and outside layer. It is used for removing noise outside the regions.
- Morphological closing: Closing first dilates the image and then, erodes the image. It is used to fill small black gaps or dots inside the white region.

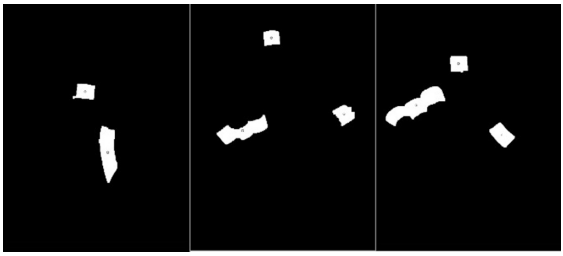


Fig. 6. Masks after noise removal (a) Red (b) Green (c) Yellow

3.4 Finding Centroids

Centroids is the arithmetic mean of all the points in the shape. They are found using moments which is the certain particular weighted average of the intensities of image pixels.

After finding out the coordinates of the centroids, the new image is shown having circles at the centroids (Fig. 7).

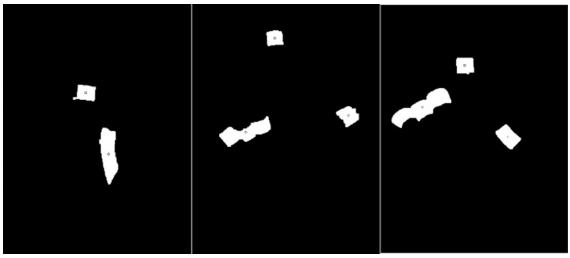


Fig. 7. Marked centroids (a) Red Mask (b) Green Mask (c) Yellow Mask

3.5 Distance Measurement

The next step is to find distances between all centroids in a particular mask Fig. (8).

- For 1 centroid, distances = 0
- For 2 centroids, distances = 1
- For 3 centroids, distances = 3
- For 4 centroids, distances = 6

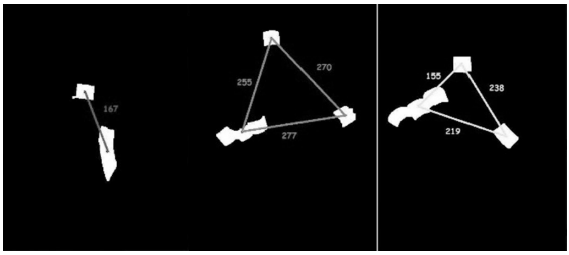


Fig. 8. Distances between centroids (a) Red (b) Green (c) Yellow

3.6 Lookup Table

Figures 9, 10 and 11 show the algorithm designed to distinguish letters on the basis of number of regions of each color. The letters having the same number of regions are then further differentiated by comparing the Euclidean distances between regions.

Here r, g and y denote the unique distances between centroids of red, green and yellow respectively when the number of regions in the masks are 2. Whereas yd1, yd2,

RED	GREEN	YELLOW	LETTER	ALGORITHM
1	1	2	E	y>175
			M	y<175
		3	N	yd1>yd3
			T	yd3>yd1
	2	2	Q	abs(g-y)<10
			E	g<100
			A	g>100
		3	P	yd1<100
			E	yd1>200 & yd2<yd3
			T	yd1>100
			S	yd1>200 & yd2>yd3
		4	E	Unique
	3	3	G	g>=100
			P	g<100
		2	G	Unique

Fig. 9. Algorithm for the case when Red Mask has 1 Region

RED	GREEN	YELLOW	LETTER	ALGORITHM
3	2	2	Q	Unique
		3	X	Unique
	3	2	B	Unique
		3	H	rd2>rd1 & yd3<yd2
			Z	rd2>rd1 & yd3>yd2
			R	rd2<rd1 & gd3>gd2 & yd1<yd2
			D	rd2<rd1 & gd3>gd2 & yd1>yd2
			Y	rd2<rd1 & gd3<gd2
	4	3	F	Unique
		4	K	Unique
4	1	3	N	Unique
	4	4	W	gd4>gd5
			K	gd4<gd5
		5	W	Unique
		6	K	Unique
5	4	4	W	Unique

Fig. 10. Algorithm for the case when Red Mask has 3, 4 or 5 regions

yd3 and similarly, gd1, gd2 and gd3 represent the distances between centroids when number of regions are 3.

3.7 Comparison

When the whole algorithm has been designed, then a test image is taken and the whole process from image acquisition to calculation of distances between centroids is done. After that, the number of regions in each mask and the distances between centroids are used to determine the letter using the algorithm formed through the lookup table (Figs. 9, 10 and 11).

RED	GREEN	YELLOW	LETTER	ALGORITHM
2	1	2	S	r>150 & abs(r-y)<70
			O	r<150 & r>100
			M	r<100 & abs(r-y)>70
	2	2	B	160<r<200 & y<200
			C	r>200 & y>200 & g>100
			O	g<100 & r>100
			Q	r<160
			J	g>180
			E	g<100 & r<100
		3	P	r<180 & yd3<yd2
			U	r<180 & yd3>yd2 & yd1<yd2
			Z	r<180 & yd3>yd2 & yd1>yd2
			I	r>250
			J	r<250 & r>200
			X	r>180 & yd2>d3
		4	R	yd1>155 & r>100
			U	yd1<155
			E	yd1>155 & r<100
	3	3	F	r>200 & gd1<100
			R	r<200 & yd2<yd1 & yd1<150
			D	r>200 & gd1>100 & yd1>yd2
			U	r>200 & gd1>100 & gd1<gd2 & yd1>yd2
			L	r<200 & gd2>gd1 & gd1>200
			Y	r<200 & gd2>gd1 & gd1<200
			G	r>200 & gd1>100 & yd1>yd2 & gd1>gd2
		4	V	gd2<gd1
			R	gd2>gd1 & yd5>yd6 & gd1<100
			Z	gd2>gd1 & yd5<yd6
			Y	gd2>gd1 & yd5>yd6 & gd1>100
		5	V	Unique
	4	3	F	Unique
		4	V	Unique

Fig. 11. Algorithm for the case when Red Mask has 2 Regions

4 Results

The total images used for training were 26 × 3 = 78. 3 images were taken for each letter. After the algorithm was developed, a total of 120 images including the training images were used for testing. The output gave 115 correct results out of 120 so the accuracy came out to be 95.8%. Figure 12 shows the confusion matrix for proposed system.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
A	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
B	0	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
C	0	2	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
D	0	0	0	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
E	0	0	0	0	8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
F	0	0	0	0	0	6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
G	0	0	0	0	0	0	6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
H	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
I	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
J	0	1	0	0	0	0	0	0	0	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
K	0	0	0	0	0	0	0	0	0	0	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
L	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M	0	0	0	0	0	0	0	0	0	0	0	0	5	0	0	0	0	0	0	0	0	0	0	0	0	0
N	0	0	0	0	0	0	0	0	0	0	0	0	0	5	0	0	0	0	0	0	0	0	0	0	0	0
O	0	0	0	0	0	0	0	0	0	0	0	0	0	0	6	0	0	0	0	0	0	0	0	0	0	0
P	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0
Q	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5	0	0	0	0	0	0	0	0	0
R	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	0	0	0	0	0	0	0	0
S	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	2	0	0	0	0	0	0	0
T	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5	0	0	0	0	0	0
U	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0
V	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5	0	0	0	0
W	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0
X	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5	0	0
Y	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5	0
Z	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5

Fig. 12. Confusion matrix

5 Conclusion

This paper presents the approach of using a black glove marked with color pointers for gesture detection. The training images are taken, separate masks are calculated for each color. Next step is to calculate centroids in the masks and then finding Euclidean distance between the regions. An algorithm is then developed based on the number of regions and the distances between centroids.

One of the main points in this approach is that the background for the images must be black or white or a varying shade of black and white or else any color that falls in the range of red, green or yellow will create problems in getting accurate results. Future work can be done in this methodology by testing the algorithm for hands of different people as this method only took data for hand of a single person. Also, further work can be done by adding data to the algorithm for another gestures e.g. numbers or basic hand gestures. But obviously, it will require a more complex algorithm to accommodate all of these gestures.

References

1. Statistics of the deaf and dumb. Sacramento Daily Union **52**(59) (1884)
2. Malenfant, H.: Introduction. In: American Sign Language: Culture, Community & Identity, chap. 1, p. 6 (2013)
3. Shukor, Z., Miskon, M.F., Jamaluddin, M.H., Ali, F.B., et al.: A new data glove approach for Malaysian sign language detection. In: IEEE International Symposium on Robotics and Intelligence Sensors (2015). Procedia Comput. Sci. **76**, 60–67 (2015)
4. Singha, J., Das, K.: Hand gesture recognition based on Karhunen-Loeve Transform. In: Mobile & Embedded Technology International Conference (2013)
5. Nachamai, M.: Alphabet recognition of American sign language: a hand gesture recognition approach using sift algorithm. Int. J. Artif. Intell. Appl. **4**(1), 105 (2013)
6. Dong, C., Leu, M.C., Yen, Z.: American sign language alphabet recognition using Microsoft Kinect. In: IEEE Conference on Computer Vision and Pattern Recognition Workshops, Boston, MA, pp. 44–52 (2015)
7. Ren, Z., Yuan, J., Meng, J., Zhang, Z.: Robust part-based hand gesture recognition using kinect sensor. IEEE Trans. Multimedia **15**(5), 1110–1120 (2013)
8. Dardas, N.H., Georganas, N.D.: Real-time gesture detection and recognition using bag-of-features and support vector machine techniques. IEEE Trans. Instrum. Measur. **60**(11), 3592–3607 (2011)
9. Bergh, M.V.D., Gool, L.V.: Combining RGB and Tof cameras for real-time 3D hand gesture interaction. In: Applications of Computer Vision, Kona, HI, pp. 66-72, January 2011
10. Ghotkar, A.S., Kharate, G.K.: Study of vision based hand gesture recognition using Indian sign language. Int. J. Smart Sens. Intell. Syst. **7**(1), 96–115 (2014)
11. Disability – A Global Picture in World Report on Disability, United Nations, WHO Press, Switzerland (2011)

A Stable Route and the Remaining Time Prediction to Send a Data Packet in Highway Environment

Mohamed Nabil[1(✉)], Abdelmajid Hajami[2], and Abdelkrim Haqiq[1]

[1] Computer, Networks, Mobility and Modeling Laboratory, FST, Hassan 1st University, Settat, Morocco
nabilmed77@gmail.com, ahaqiq@gmail.com
[2] LAVETE Laboratory, FST, Hassan 1st University, Settat, Morocco
abdelmajidhajami@gmail.com

Abstract. There are numerous research challenges to develop safety or non-safety applications in vehicular ad hoc networks (VANETs). One of the critical issues consists in designing a scheme that solves the local maximum problem, reduces contention phase, increases the route lifetime and reduces the frequent path disruptions caused by high mobility of vehicles. Existing schemes do not take into consideration all these issues. For this reason, this paper addresses these issues for non-safety applications in a highway environment. We propose a scheme that strives to find a stable route, minimize contention phase and predict the route lifetime and the average transmission time of the data packets. Our scheme increases the percentage of packets delivery, reduces the control overhead and decreases number of error messages generated during transmission of data packets as function of vehicles density.

Keywords: Stable route · Prediction the route lifetime · Stable neighbor · IDM_LC · Highway scenario · VANET · NS2

1 Introduction

A Vehicular Ad hoc Network (VANETs) is a case of MANET where nodes are vehicles. VANETs provide a wireless communication between vehicles, using a dedicated short range communication. Each vehicle can communicate with other vehicles directly through the device On Board Unit (OBU) forming vehicle to vehicle communication (V2 V) or communicate with fixed equipment beside the road, referred to as Road Side Unit (RSU) forming vehicle to infrastructure communication (V2I). These types of communications allow vehicles to share different sorts of information. The aim behind sharing this information is to provide a safety message to warn drivers about expected hazards in order to decrease the number of accidents by enabling a set of safety applications and to provide passengers comfort by enabling a set of non-safety applications. These applications can provide drivers or passengers with weather and traffic information and detail the location of the nearest restaurant, petrol station or hotel. They can allow passengers to play online games, access the internet and check their emails while the vehicle is connected to the infrastructure network [1].

A. Abraham et al. (eds.), *Proceedings of the 16th International Conference on Hybrid Intelligent Systems (HIS 2016)*, Advances in Intelligent Systems and Computing 552,
DOI 10.1007/978-3-319-52941-7_2

Several challenges await researchers to develop these applications. Among these challenges is to design an efficient routing protocol that can increase the route lifetime duration, deliver a packet in a minimum period of time and be suitable for high density of vehicles. Hence, this protocol will increase the percentage of packets delivery with few dropped packets and will reduce the control overhead and the increasing throughput.

Numerous schemes were proposed to address this issue. Among them the ones are based on the reactive approach in which the next forwarder is chose on real time. Schemes of this approach aim to reduce the number of hops, to increase the route lifetime and to select the stable route and the shortest distance path between source and destination. A contention phase will be created whenever selecting the next forwarder. This contention phase introduces few units of delay which will be substantial in multi-hop communication in dense environment. Selecting the stable routes can lead to the local maximum problem. These schemes still suffer from frequent breaking of the route during the data transmission. Hence, several data packets will be lost.

To overcome these limitations, we propose a routing protocol through which the choice of the route is based on selecting the most stable neighbor with the transmitter of route request message giving priority to neighbor that travels in the same direction of the transmitter movement. This protocol also allows predicting the lifetime of the route and the transmission average time of the data packets; hence, it allows predicting the left time of route to send a data packet to minimize the number of errors and the number of the lost data packets. To minimize contention phase, each vehicle periodically determines two most stable neighbors (one of them is in the front and the other is in the back of vehicle) as well as their stability time.

The rest of the paper is organized as follows. Section 2 presents related work. Section 3 shows our scheme. Section 4 presents simulation and results. Finally, we give a conclusion in Sect. 5.

2 Related Work and Motivations

One of the well-known schemes is GPSR [2]; it uses the greedy forwarding method whereby next hop is chosen based on node that is closer to the destination node. When the local maximum problem happens, GPSR switches to a perimeter routing algorithm. In a highway environment, this protocol does not select the most stable route that decreases the frequency of breaking route; hence the increase of the number of error messages because it does not take in consideration the route stability at forwarding packets. The authors in [3] studied LAR [4] in highway scenario. The protocol was tested against vehicles density for various metric (throughput, packets delivery ratio, end to end delay, and overload) with a high speed. The protocol has good performance in a communication environment and it is sensitive to the density of vehicles and the number of lanes. D-LAR [5, 6] is a greedy approach that combines LAR with DIR [7] to forward packets in the request zone to the direct neighbor having direction closest to the line drawn between source and destination. This protocol does not take into consideration the direction of movement of the forwarder vehicle. Hence, it cannot choose the most stable route in the case where there are two neighbors of the sender, one of them travels in opposite direction and it is the closest to the line drawn between source and

destination. MOPR [8] uses the moving information of vehicles to predict future positions of vehicles and to estimate the time needed for the transmission of data to decide whether a route is likely to be broken or not during the transmission time. MOPR allows avoiding link broken by high mobility of vehicles during data transmission. The performance of this algorithm depends on the used transmission protocols. In [9], the authors have grouped vehicles according to their direction of movement. The stability of the communication is ensured by the choice of the most stable path using the ROMSGP scheme. This choice is made based on the calculation of LET of each path. The longest LET path is considered to be the most stable. The authors did not take into consideration the case where vehicles travel in opposite direction if there is no vehicle travelling in the same direction of group movement. RBVT-R [10] is a reactive source routing protocol for VANETs that creates routes on demand by using "connected" road segments. A connected road segment is a segment between two adjacent intersections with enough vehicular traffic to ensure network connectivity. When a node receives a new Route Discovery (RD), it holds the packet for a period of time inversely proportional to the distance between itself and the sending node. After the waiting period, a node rebroadcasts the RD packet only if it did not notice that this packet was rebroadcast by nodes that are located further on the same road segment. The waiting period becomes substantial in multi-hop communication in highway. Besides, the protocol not takes into count the route stability because the RD packet will be rebroadcasted by the vehicles that are located further on the same road segment. SCRP [11] is a distributed routing protocol that builds stable backbones on road segments using connected dominating sets (CDS). The vehicle with the lowest SF (vehicle's stability factor) is added to the backbone. The latter chooses the neighbor with the lowest SF as the next forwarder; hence, it becomes a backbone vehicle. Then, it selects the next vehicle to be included in the backbone. This procedure is repeated until the all of road segment is covered. In highway environment, this scheme builds one path only on long of road in a proactive manner. Hence, all source vehicles share the same path or part of path. While the number of source vehicles increases, the performance of protocol decreases (throughput, packets delivery ratio, end to end delay).

Besides of these above issues of these routing protocols, most of them are focused to improve the urban environment. They do not take into count the highway environment which characterized by high speed vehicle mobility. This constraint makes routing very challenging in highway. Our scheme strives to minimize contention phase, avoid route break before of its apparition, by predicting the remaining time of route to send a data packet. Also, it allows finding stable route by selecting the most stable neighbor of the transmitter of route request message giving priority to the neighbor that travels in the same direction of the movement of this transmitter.

3 Stable Route and Remaining Time Prediction (SRRTP)

SRRTP is designed for non-safety applications in highway environment. It is on demand routing protocol, it is similar to LAR [4] and AODV [12] protocols.

The network model consists of one road ended by two intersections in highway environment or in urban environment for roads segments. This road has the same

characteristics such as length, width, number of lanes. Each lane has a distinctive traffic density (see Fig. 1). Each vehicle is equipped with a global positioning system (GPS) that provides information about its location, speed, and direction. Finally, each source node knows the location of the destination by using a location service such as RLSMP [13].

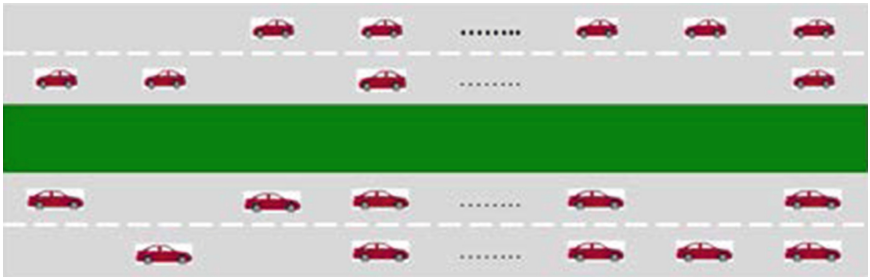


Fig. 1. Bi-directional highway model

SRRTP is based on the following considerations: most-stable neighbor, minimization contention phase, route lifetime prediction, transmission time prediction of the data packet and route construction.

3.1 Most-Stable Neighbor

This section is presented in our work [14] and extended in our works [15, 16]. There are four cases to calculate the time (t) of each neighbor.

First case: The vehicles I and A have the same direction of movement and do not have the same velocity at time t0 [15, 16]:

$$t = \frac{X_I - X_A}{V_A - V_I} + \frac{\sqrt{R^2 - (Y_I - Y_A)^2}}{|V_A - V_I|} \tag{1}$$

Second case: The vehicles I and A have the same direction of movement and they have the same velocity at time t0 [16]: In this case, the transmitting vehicle calculates the distance d between itself and each neighbor which has the same speed by this formula:

$$d(A, I) = \sqrt{(X_A - X_I)^2 + (Y_A - Y_I)^2} \tag{2}$$

The vehicle which has the closest distance from the R/2 will receive and transmit the route request message.

Third case: The forwarding vehicle has an opposite direction of its neighbor at time t0 and they are approaching each other [11].

$$t = \frac{R + d(A, I)}{|V_A + V_I|} \tag{3}$$

Fourth case: The forwarding vehicle has an opposite direction of its neighbor at time t0 and they are moving away from each other [11].

$$t = \frac{R - d(A, I)}{|V_A + V_I|} \tag{4}$$

3.2 Minimization of Contention Phase

If each vehicle which has a route request message will perform operations to select the vehicle that will receive and retransmit the message, the accumulation time of these operations of all participating vehicles in route becomes substantial. Therefore, these operations increase the route request time between sources and destinations. Hence, decreasing the route lifetime at transmission of data packet and a new route request message can be triggered in entire network. To avoid this contention phase, each node periodically determines the most stable neighbor onward and the most stable neighbor backward; at the same time, it determines the time of stability of each of these neighbors. Each vehicle seeks these most stable neighbors (onward and backward) among those traveling in the same direction of its movement. If there are no neighbors that have the same direction of its motion, then it searches among those traveling in the opposite direction. Hence, we give priority to the neighbors that travel in the same direction of the transmitter movement to receive and forward the route request message.

The vehicle which has the route request message chooses (among its neighbors in the half-circle of its coverage area in the side closing to the destination) that has the longest time for receive and forward this message.

3.3 Route Lifetime Prediction

To predict the lifetime of the route established between sources and destinations, each source adds the stability time (TS) (it is calculated and determined in Sects. 3.1 and 3.2) of the next transmitter relative to the current transmitter in the route request packet. Each vehicle receives this route request packet compares the stability time of the next hop to the one in the request packet. The shortest stability time will put in the route request packet instead of the other.

TS = min (E ($TS_{i,j}$)) where E is the set of links between vehicles that build the route between source and destination. E ($TS_{i,j}$) is the set of lifetimes of these links between i and j vehicles that build the route between source and destination.

When the destination receives the route discovery packet, it copies the TS, its current location, its current speed and current time in the route reply packet and it sends this packet back downstream at source.

3.4 Transmission Time Prediction of the Data Packet

Transmission time of a data packet between the source and destination get changed according to the density of vehicles in highway (see Fig. 2). Therefore, it is hard to predict a constant value during the time of the simulation. To solve this problem, each time the destination receives a data packet, calculates and stores the average time of data packet transmission. When a route request reaches to the destination, the latter sends the average time in the route reply message that will be sent back downstream to source. The latter calculates the remaining time of route between itself and the destination, if it wants to send a data packet. If time left of route is less than the average transmission time of a data packet, the source launches a new route request; otherwise, it sends the data packet.

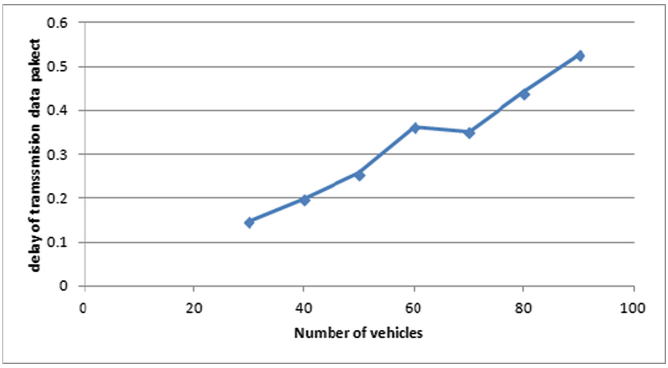


Fig. 2. Average transmission time of a data packet versus vehicles density

3.5 Route Construction

When the source S wants to send a data packet to the destination D, it checks its routing table. If there is a route to the destination (enabled route), S calculates the remaining time of route between itself and the destination.

If this time of route is strictly greater than the average transmission time of a data packet, the source sends the data packet; otherwise, S checks its list of neighbors to the destination. If D is listed then S updates the route to the destination and sends the data packet; otherwise, S generates a new route request message. If D is located towards the direction of S movement, S sends the route request to the neighbor that is onward (the neighbor onward that remains the longest time in the S coverage area); otherwise, S sends the route request to the neighbor that is backward (the neighbor backward that remains the longest time in the S coverage area). Each vehicle (I) receives a route request message it will forward the message to its onward vehicle if it received the message from its back vehicle, otherwise it will forward the message to its back vehicle. This operation will be repeated until arrival to destination. When the destination receives the route discovery packet, it copies the stability time (TS), average time of data packet

transmission, its current location, its current speed and current time of speed in the route reply message and sends it downstream toward the source.

4 Simulation and Results

We have used the pattern IDM-LC that is a microscopic mobility model in the tool Vehicular Ad Hoc Networks Mobility Simulator (VanetMobiSim) [17, 18] and we have used NS2 [19] to implement our protocol. Vehicles are deployed in a 4000 m × 100 m area. This area is a highway with four lanes bidirectional; its ends are set by traffic lights. Vehicles are able to communicate with each other using the IEEE 802.11 MAC layer. The vehicles' speed fluctuates between 0 m/s and 27 m/s. We have considered packet size of 512 bytes, Simulation Time of 400 s, hello interval of 1 s and packet rate of 4 packets per second. We setup ten multi-hop CBR flow vehicles over the network that start at different time instances and continue throughout the remaining time of the simulation. The transmission range is kept at 250 m. Simulation results are averaged over 20 simulation runs. Location-Aided Routing (LAR1) is used to compare it with our protocol; because we developed this protocol from source code of LAR1. These protocols are evaluated for packet delivery ratio, normalized routing load, number of generated errors and number of errors received at source according to vehicles density.

Packet delivery ratio: Figure 3 shows that our scheme has good packet delivery ratio and it clearly outperforms LAR1. This is because our scheme forwards data packets over roads by predicting both the route lifetime and the time of transmission of the data packet. Also, it chooses a stable route. But with the increase in network density, the packet delivery ratio of the two schemes decreases. This is because when network density increases, the transmission time of data packets increases; and also we do not yet use a method such as carry-and-forward to recuperate the lost data packets.

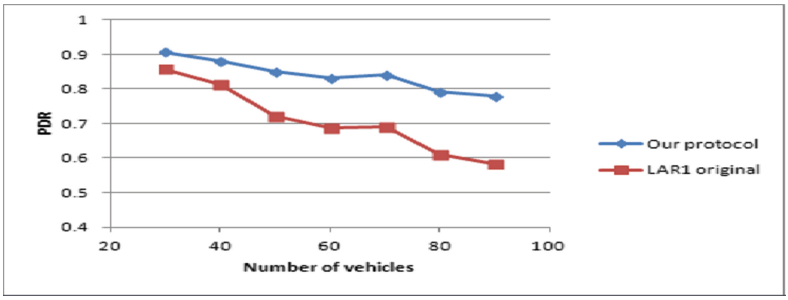


Fig. 3. PDR as a function of vehicle density

Normalized routing load: Figure 4 shows that Normalized Routing Load increases with increasing the density of network. Our scheme has a lowest normalized routing load compared to LAR1. This is explained by the decrease of route discovery process in reason of the select the stable route and the predicting the remaining time of route.

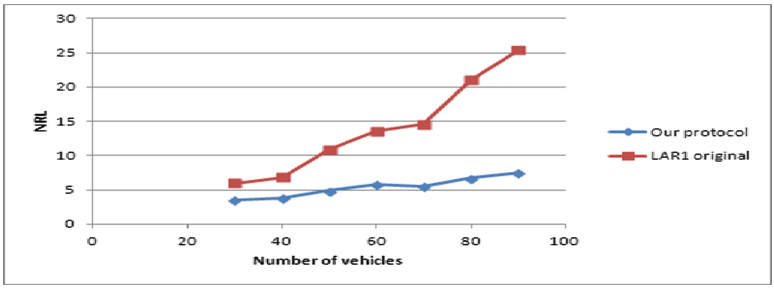


Fig. 4. NRL as a function of vehicle density

Number of error messages generated during transmission of data packets: Fig. 5 shows that the number of errors increases with increasing the density of the network. Our scheme has the lowest number of errors compared to LAR1. This is explained by predicting the route lifetime and the time of transmission of the data packet. Therefore, our protocol calculates the time left of route to the destination. So, it takes the decision before sending data packets.

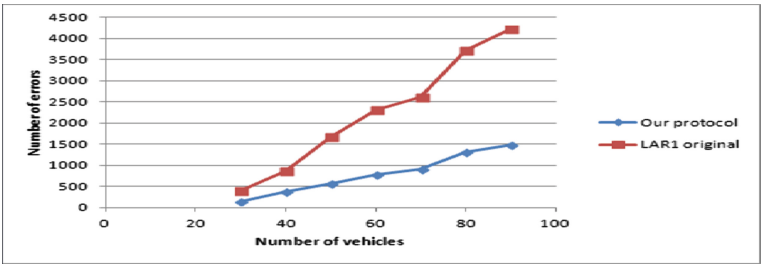


Fig. 5. Number of errors as a function of vehicle density

Number of error messages received at source: Fig. 6 shows the number of error messages received at source with varying network density. The percentage of error messages that arrived at source is always over 70% compared to that of LAR1. This is because of the stability of our protocol.

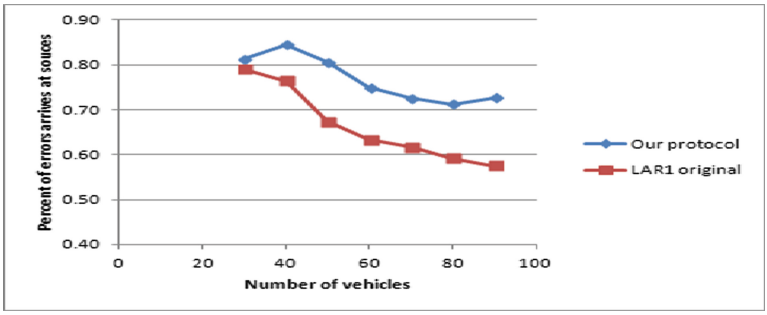


Fig. 6. Number of errors received at source as a function of vehicle density

5 Conclusion

Our protocol is designed to improve the communication in vehicular ad hoc networks in highway scenarios for non-safety applications. It strives to find a stable route, minimize contention phase and predict the time left of route to the destination before sending data packets to avoid the route disruption prior to its happening. Our protocol is based on four considerations that are stable route, minimization of contention phase, route lifetime prediction and transmission time prediction of the data packet. Our scheme increases the percentage of packets delivery, reduces the control overhead and decreases number of error messages generated during transmission of data packets. It is evaluated as function of vehicle density and it is compared with LAR scheme1. It is found that it outperforms LAR1 original in highway environment by using IDM_LC to generate realistic mobility files.

For future work, we will study the case of eliminating of error messages. Because, the source vehicle is able to take the decision about the transmission of data packet by calculating the time left of route to destination prior to send data packets.

References

1. Al-Sultan, S., Al-Doori, M.M., Al-Bayatti, A.H., Zedan, H.: A comprehensive survey on vehicular ad hoc network. J. Netw. Comput. Appl. **37**, 380–392 (2014)
2. Karp, B., Kung, H.T.: GPSR: greedy perimeter stateless routing for wireless networks. In: Proceedings of IEEE/ACM MobiCom, Boston, MA, pp. 243–254, August 2000
3. Husain, A., Kumar, B., Doegar, A.: A study of Location Aided Routing (LAR) protocol for vehicular ad hoc networks in highway scenario. Int. J. Eng. Inf. Technol. **2**(2), 118–124 (2010)
4. Ko, Y.-B., Vaidya, N.H.: Location aided routing in mobile ad hoc networks. ACM J. Wirel. Netw. **6**(4), 307–321 (2000)
5. Raw, R.S., Das, S., Singh, N., Kumar, S., et al.: Feasibility evaluation of VANET using directional-location aided routing (D-LAR) protocol. Int. J. Comput. Sci. Issues **9**(5), 404–410 (2012)

6. Raw, R.S., Lobiyal, D.K., Das, S., Kumar, S.: Analytical evaluation of directional-location aided routing protocol for VANETs. Wirel. Pers. Commun. **82**(3), 1877–1891 (2015)
7. Stojmenovic, I., Ruhil, A.P., Lobiyal, D.K.: Voronoi diagram and convex hull based geocasting and routing in wireless networks. Wirel. Commun. Mob. Comput. **6**(2), 247–258 (2006). Special Issue on Ad Hoc Wireless Networks, John Wiley & Sons Ltd
8. Menouar, H., Lenardi, M., Filali, F.: A movement prediction based routing protocol for vehicle-to-vehicle communications. In: Proceedings of V2VCOM, San Diego, CA, July 2005
9. Taleb, T., Sakhaee, E., Jamalipour, A., Hashimoto, K., Kato, N., Nemoto, Y.: A stable routing protocol to support ITS services in VANET networks. IEEE Trans. Veh. Technol. **56** (6), 3337–3347 (2007)
10. Nzouonta, J., Rajgure, N., Wang, G., Borcea, C.: VANET routing on city roads using real-time vehicular traffic information. IEEE Trans. Veh. Technol. **58**(7), 3609–3626 (2009)
11. Togou, M.A., Hafid, A., Khoukhi, L.: SCRP: stable CDS-based routing protocol for urban vehicular ad hoc networks. IEEE Trans. Intell. Transp. Syst. **17**(5), 1298–1307 (2016)
12. Perkins, C., Belding-Royer, E., Das, S.: Ad hoc On-Demand Distance Vector (AODV) routing. RFC3561, IETF MANET Working Group, July 2003
13. Saleet, H., Basir, O., Langar, R., Boutaba, R.: Region-based location-service-management protocol for VANETs. IEEE Trans. Veh. Technol. **59**(2), 917–931 (2010)
14. Nabil, M., Hajami, A., Haqiq, A.: Improvement of location aided routing protocol in Vehicular Ad Hoc Networks on highway. In: 2015 5th World Congress on Information and Communication Technologies (WICT), Marrakech, Morocco, pp. 53–58 (2015)
15. Nabil, M., Hajami, A., Haqiq, A.: Improvement of route lifetime of LAR protocol for VANET in highway scenario. In: 2015 IEEE/ACS 12th International Conference of Computer Systems and Applications (AICCSA), Marrakech, Morocco, pp. 1–8 (2015)
16. Nabil, M., Hajami, A., Haqiq, A.: Increasing the route lifetime stability of LAR protocol for VANETs in highway environment. J. Netw. Innov. Comput. **4**, 001–010 (2016). ISSN 2160–2174
17. Harri, J., Fiore, M.: VanetMobiSim- Vehicular Ad Hoc Network Mobility Extension to the Canumobisim Framework, Manual. Institut Eurecom/Politecnico di Torino, Italy (2006)
18. Fiore, M., Harri, J., Filali, F., Bonnet, C.: Vehicular mobility simulation for VANETs. In: Proceedings of the 40th Annual Simulation Symposium (ANSS 2007), pp. 301–309. IEEE Computer Society, Washington (2007)
19. NS-2 Mannual. http://www.isi.edu/nsnam/ns/ns-documentation.html

Complex-Valued Neural Network Model and Its Application to Stock Prediction

Haifeng Wang, Bin Yang(✉), and Jiaguo Lv

School of Information Science and Engineering,
Zaozhuang University, Zaozhuang 277160, China
batsi@l26.com

Abstract. In this paper, a novel complex-valued neural network (CVNN) algorithm is proposed to predict stock index. In a CVNN, input layer, weights, threshold values and output layer are all complex numbers. Cuckoo search (CS) is proposed to optimize the complex parameters. NIFTY stock market indices and Shanghai stock exchange composite index are used to evaluate the performance of CVNN. The results reveal that CVNN performs better than the classical real neural networks.

Keywords: Complex-valued · Neural network · Cuckoo search · Stock index · Prediction

1 Introduction

A stock index (stock market index) measures the value of stock for some time in the economy or financial markets, which could track economic health from many perspectives and the performance of powerful companies in order to forecast economic trends. However it is influenced by many intricately related economic and political factors, so it is a nonlinearity, nonstationary and complex process [1]. Forecasting stock index is one of the most profitable and challenging tasks for researchers [2].

In recent years, many new methods for forecasting the stock index have been developed. Artificial neural networks (ANN) with intrinsic robustness and nonlinear characteristics could approximate any continuous function and have been widely used to model stock market time series [3, 4]. Zhang et al. [5] proposed an improved bacterial chemotaxis optimization and back propagation (BP) neural network for prediction of various stock indices. Fang et al. [6] proposed a stock market prediction model based on genetic algorithms (GA) and wavelet neural networks (WNN) and reported significantly better accuracies compared to existing approaches. Chen proposed hierarchical radial basis function (HRBF) Networks and flexible neural tree model (FNT) to forecast financial time series data and evaluated the prediction performance of the flexible neural tree algorithm, wavelet neural network, local linear wavelet neural network, and feed-forward artificial neural network.

Recently, complex-valued neural network (CVNN) has been proposed to predict the time series data. Compared with real neural network, CVNN is more flexible and functional. Xiong et al. proposed the fully complex-valued radial basis function neural

A. Abraham et al. (eds.), *Proceedings of the 16th International Conference on Hybrid Intelligent Systems (HIS 2016)*, Advances in Intelligent Systems and Computing 552,
DOI 10.1007/978-3-319-52941-7_3

networks (FCRBFNNs) to investigate the possibility of forecasting interval time series [9]. Kitajima et al. proposed a wind speed prediction system using complex-valued neural network and frequency component of observed wind speed data as an input information [10]. Aizenberg et al. proposed Multilayer Neural Network with Multi-Valued Neurons to forecast oil production [11].

In this paper, we propose complex-valued neural network to predict stock index. CVNN has complex valued inputs/outputs, weights and activation functions. By analysis of some evolutionary algorithms, CS is potentially far more efficient than particle swarm optimization (PSO), genetic algorithms (GA) and artificial bee colony (ABC) [13]. So cuckoo search is proposed to optimize the complex parameters (real and imaginary parts) and real parameters.

2 Method

2.1 Complex-Valued Neural Network

In a complex-valued neural network, input layer, weights, threshold values and output layer are all complex numbers. A three-layer complex-valued neural network is described in Fig. 1, which has m input nodes, n hidden nodes and one output node. Suppose that input vector $[z_1, z_2, \cdots, z_m]$. The result of i-th hidden node is computed as followed.

$$h_i = f(W_i + W_{i1}z_1 + W_{i2}z_2 + \ldots + W_{im}z_m). \tag{1}$$

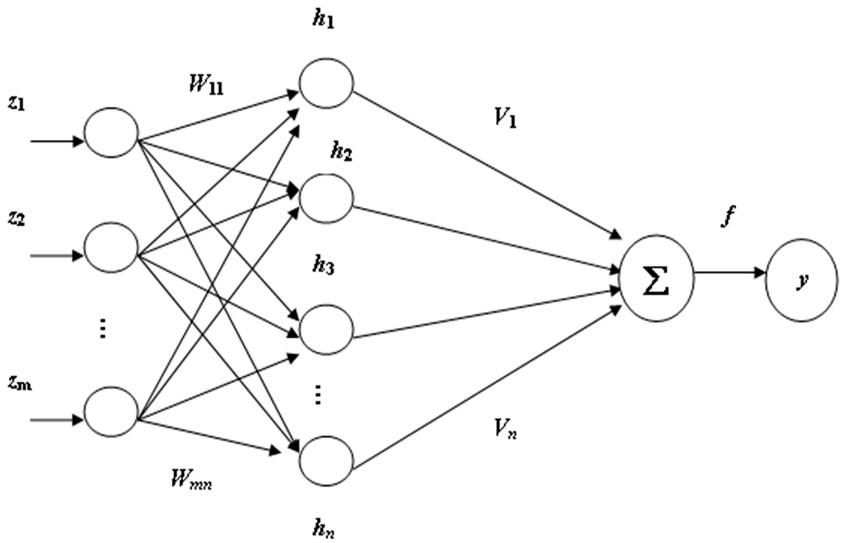


Fig. 1. A three-layer complex-valued neural network

Where W_i is the threshold value of i-th hidden node and $W_{i1}, W_{i2}, \ldots W_{im}$ are the weights. f is the activation function, which is:

$$f(z) = \frac{z}{c + \frac{1}{r}|z|}. \tag{2}$$

Where c and r are the real variables.

The output layer is computed:

$$y = f(h_1 V_1 + h_2 V_2 + \ldots + h_n V_n). \tag{3}$$

2.2 Optimization of Parameters

To find the optimal parameters of CVNN, the cuckoo search (CS) method is used below. CS is one of the latest nature-inspired metaheuristic algorithms, proposed in 2009 by Yang [12].

CS follows three idealized rules: (1) Each cuckoo only lays one egg at a time, which is dumped in randomly selected nest; (2) The nests with high quality of eggs could be selected to the next generation; (3) The number of available nests is fixed, and the egg laid by other cuckoo is discovered by the host cuckoo with a probability $p_a \in [0, 1]$.

According to the above three idealized rules, the process of CS is described as followed.

(1) In a CVNN (Fig. 1), many parameters need to be optimized, containing weights (W_{ij} and V_i), threshold value (W_i) and parameters in activation function (c and r). Due to the fact that weights and threshold value are complex numbers, both real and imaginary parts need to be optimized. For a CVNN m-n-1 (m inputs, n hidden neurons and 1 output), the number of parameters is $2\ nm + 6n + 2$ and parameter vector is $[\mathrm{Re}(W_{11}), \mathrm{Im}(W_{11}), \ldots, \mathrm{Re}(W_{mn}), \mathrm{Im}(W_{mn}), \mathrm{Re}(W_1), \mathrm{Im}(W_1), \ldots, \mathrm{Re}(W_n), \mathrm{Im}(W_n), c_1, r_1, \ldots, c_n, r_n, \mathrm{Re}(V_1), \mathrm{Im}(V_1), \ldots, \mathrm{Re}(V_n), \mathrm{Im}(V_n)]$. Create the initial population randomly $X_i (i = 1, 2, \ldots q)$, which represent host nests.

(2) Compute the fitness values of all population. If the maximum number of generations is reached or a satisfactory solution is found, then stop.

(3) Lévy flight is performed to generate new solutions.

$$X_i^{t+1} = X_i^t + \alpha \oplus L\acute{e}vy(\lambda). \tag{4}$$

Where X_i^t is the i-th solution at t-th generation, α is step size which could control the scale of random search. In general, $\alpha = 1$. $\oplus$ means entrywise multiplications, $L\acute{e}vy(\lambda)$ abides by Lévy probability distribution:

$$Lévy(\lambda) \;\; u = t^{-1-\lambda}, \;\; 0 < \lambda \leq 2. \tag{5}$$

$Lévy(\lambda)$ could be computed using the following Eq. [13]:

$$\begin{aligned} Lévy(\lambda) \;\; &= \frac{\phi \times \mu}{|v|^{\frac{1}{\lambda}}} \\ \phi &= \left(\frac{\Gamma(1+\lambda) \times \sin(\Pi \times \frac{\lambda}{2})}{\Gamma((\frac{1+\lambda}{2}) \times \lambda \times 2^{\frac{\lambda-1}{2}})}\right)^{\frac{1}{\lambda}}. \end{aligned} \tag{6}$$

Where μ and v follow Gaussian distributions.

(4) According to predefined probability p_a, discard the worse solutions. Create the same number of new solutions using preference random walk.

$$X_i^{t+1} = X_i^t + r(X_m^t - X_n^t). \tag{7}$$

Where r is scaling factor, which is created randomly from [0, 1]. X_m^t and X_n^t are random solutions at t-th generation. Go to step 2).

2.3 Flowchart of Method

(1) The stock index is real number, so input and output data need be transformed to complex values before being inputted into CVNN model. The following transformation steps are used [11]:

- Suppose the input real numbers $[x_1, x_2, \ldots, x_m]$, and tally the maximum and minimum values of real numbers (*max* and *min*).
- i-th real number x_i is transformed into complex number as followed.

$$\begin{aligned} \varphi_i &= \frac{x_i - min}{max - min}(2\pi - \delta), \\ z_i &= e^{i\varphi_i}. \end{aligned} \tag{8}$$

Where δ is the shift angle and i stands for the value of $\sqrt{-1}$.

2) Train the complex-valued neural network according to the above complex numbers (input data) and cuckoo search, which is introduced in detail in Sect. 2.2. The output of CVNN is complex number, so the output value needs to be transformed into real number in order to evaluate model effectively. The inverse transformation shall be used:

$$\begin{aligned} &\arg z = \varphi, \\ &y = \frac{\varphi(max - min)}{2\pi - \delta} + min. \end{aligned} \tag{9}$$

Where $\arg z$ is the argument of complex value z.

3 Experiment

3.1 Data Sets and Evaluation Criteria

We choose the NIFTY index from 01 January 1998 to 03 December 2001 and the Shanghai stock exchange composite index (Shanghai index) from 04 January, 2011 to 01 January, 2015 as the samples for validating the model. Each of the samples is composed of opening price, closing price, low price, high price and trading volume. The root mean squared error (*RMSE*), maximum absolute percentage error (*MAP*), mean absolute percentage error (*MAPE*) are used to test the performance of the method. Three criteria are defined as followed.

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(f^i_{target} - f^i_{forecast}\right)^2} \tag{10}$$

$$MAP = \max\left(\frac{|f^i_{target} - f^i_{forecast}|}{f^i_{forecast}} \times 100\right) \tag{11}$$

$$MAPE = \frac{1}{N}\sum_{i=1}^{N}\left(\frac{|f^i_{target} - f^i_{forecast}|}{f^i_{forecast}}\right) \times 100 \tag{12}$$

Where N represents the total time points, f^i_{target} is the actual index value on day i and $f^i_{forecast}$ is the forecasting index value on day i.

3.2 Forecasting Results

For NIFTY index, 400 samples are used to train CVNN and 384 samples are used to test performance. For Shanghai index, 500 and 244 samples are used to train and test, respectively. The parameters in this experiment are chosen experientially and given in Table 1. To evaluate the performance of CVNN, we compare it with several classic methods including NN-PSO, WNN-PSO, FNT from Chen and Abraham [8].

Through 20 runs, the averaged results are listed in Tables 2 and 3. Results show that CVNN model has the smallest RMSE, Map and MAPE and is an effective model for estimating one day ahead stock prices compared to other advanced neural networks. The results of two stock data set predictions are shown in Figs. 2 and 3, which depict

Table 1. Parameters used in this experiment

Parameters	Initial value
Population size in CS	50
Maximum steps	100
p_a in CS	0.3
Shift angle δ	π

Table 2. Comparison results among four different methods for NIFTY index

	RMSE	MAP	MAPE
NN-PSO [8]	0.01326	27.257	6.347
WNN-PSO [8]	0.01426	39.671	7.002
FNT [8]	0.01328	27.028	6.365
CVNN	**0.01303**	**26.865**	**6.215**

Table 3. Comparison results among four different methods for Shanghai index

	RMSE	MAP	MAPE
NN-PSO [8]	0.01275	45.865	7.092
WNN-PSO [8]	0.01191	35.778	6.4279
FNT [8]	0.01182	27.688	6.2696
CVNN	**0.01035**	**20.1088**	**5.3325**

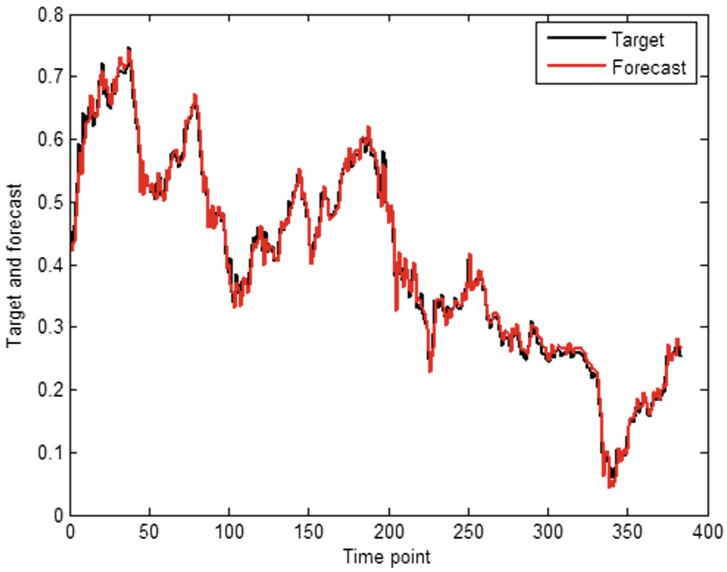


Fig. 2. The test results for the one day ahead prediction of the the NIFTY index

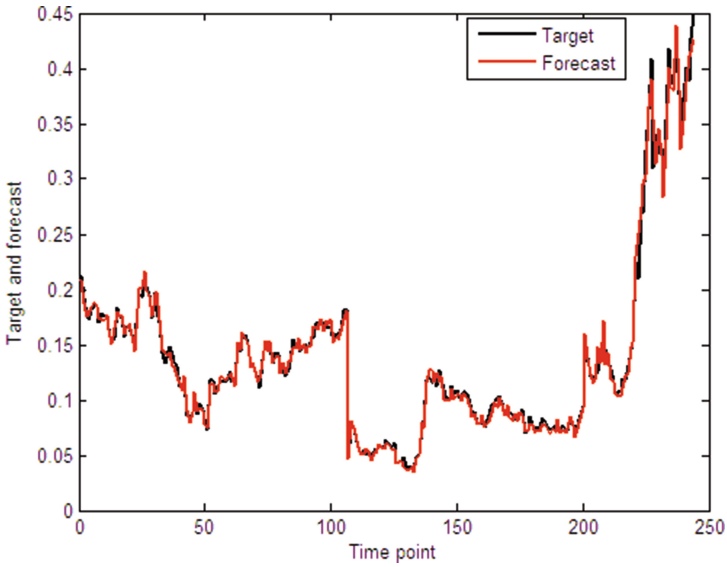


Fig. 3. The test results for the one day ahead prediction of the Shanghai index

the one day future actual stock value and predicted stock value. It could be seen that CVNN model could forecast the one day ahead stock index well and provide appropriate generalization.

4 Conclusion

In this paper, a novel approach based on complex-valued neural network (CVNN) model is proposed to forecast stock market. Cuckoo search is proposed to optimize the complex parameters and real parameters. NIFTY stock market indices and Shanghai stock exchange composite index are applied to test our model. Compared with the NN, WNN and FNT, it is concluded that CVNN model can outperform all other popular models.

The number of parameters optimized is large, so optimization process spent more time. In the future, we plan to improve evolutionary algorithm in order to increase efficiency.

Acknowledgements. This study was funded by the Ph.D. research startup foundation of Zaozhuang University (No. 2014BS13), foundation of Zaozhuang University (2015YY02) and Shandong Provincial Natural Science Foundation, China (No. ZR2015PF007).

References

1. Wei, L.Y.: A hybrid ANFIS model based on empirical mode decomposition for stock time series forecasting. Appl. Soft Comput. **42**, 368–376 (2016)
2. Sadaei, H.J., Enayatifar, R., Lee, M.H., Mahmud, M.: A hybrid model based on differential fuzzy logic relationships and imperialist competitive algorithm for stock market forecasting. Appl. Soft Comput. **40**, 132–149 (2015)
3. Angeline, P.J., Saunders, G.M., Pollack, J.B.: An evolutionary algorithm that constructs recurrent neural networks. IEEE Trans. Neural Netw. **5**, 54–65 (1994)
4. Al-Askar, H., Hussain, A.J., Al-Jumeily, D., Radi, N.: Regularized dynamic self organized neural network inspired by the immune algorithm for financial time series prediction. In: Huang, D.-S., Han, K., Gromiha, M. (eds.) ICIC 2014. LNCS, vol. 8590, pp. 56–62. Springer, Heidelberg (2014). doi:10.1007/978-3-319-09330-7_8
5. Zhang, Y.D., Wu, L.N.: Stock market prediction of S&P 500 via combination of improved BCO approach and BP neural network. Expert Syst. Appl. **36**, 8849–8854 (2009)
6. Fang, Y., Fataliyev, K., Wang, L., Fu, X., Wang, Y.: Improving the genetic-algorithm-optimized wavelet neural network for stock market prediction. In: International Joint Conference on Neural Networks, pp. 3038–3042 (2014)
7. Chen, Y., Peng, L., Abraham, A.: Stock index modeling using hierarchical radial basis function networks. In: Gabrys, B., Howlett, R.J., Jain, L.C. (eds.) KES 2006. LNCS (LNAI), vol. 4253, pp. 398–405. Springer, Heidelberg (2006). doi:10.1007/11893011_51
8. Chen, Y.H., Abraham, A.: Hybrid learning methods forstock index modeling. In: Kamruzzaman, J., Begg, R.K., Sarker, R.A. (eds.) Artificial Neural Networks in Finance, Health and Manufacturing: Potential and Challenges. IdeaGroup Inc. Publishers, Hershey (2006)
9. Xiong, T., Bao, Y., Hu, Z., Chiong, R.: Forecasting interval time series using a fully complex-valued RBF neural network with DPSO and PSO algorithms. Inf. Sci. **305**, 77–92 (2015)
10. Kitajima, T., Yasuno, T., Ikeda, N.: Wind speed prediction system using complex-valued neural network and frequency component of wind speed. IEICE Tech. Rep. Neurocomput. **113**, 35–40 (2013)
11. Aizenberg, I., Sheremetov, L., Villa-Vargas, L.: Multilayer neural network with multi-valued neurons in time series forecasting of oil production. Neurocomputing **8495**, 61–70 (2015)
12. Yang, X.S., Deb, S.: Cuckoo search: recent advances and applications. Neural Comput. Appl. **24**(1), 169–174 (2014)
13. Civicioglu, P., Besdok, E.: A conceptual comparison of the cuckoo-search, particle swarm optimization, differential evolution and artificial bee colony algorithms. Artif. Intell. Rev. **39** (4), 315–346 (2013)

Separation of Vertebrae Regions from Cervical Radiographs Using Inter-Vertebra Distance and Orientation

Anum Mehmood, M. Usman Akram(✉), Mahmood Akhtar, and Anam Usman

College of Electrical and Mechanical Engineering,
National University of Sciences and Technology, Islamabad, Pakistan
anummehmood81@ce.ceme.edu.pk, usmakram@gmail.com,
mahmood.akhtar@ceme.nust.edu.pk

Abstract. For many orthopedics, neurosurgeon and radiologists, the extraction of the spinal column and detection of each vertebra is essential. It helps in the identification of vertebral abnormalities like cervical trauma, osteoporosis, spinal ruptures etc. Computer aided diagnostic systems for the localization and extraction of vertebra from X-ray images are important to perform mass screening. It is a challenging task to be performed due to low contrast imaging and noise present in X-ray images. In this paper, we present a technique for automatic detection and extraction of vertebrae area. The proposed technique takes a radiograph as input and detects the location of cervical vertebrae (C_3–C_7) using generalized Hough Transform and Fuzzy C-mean clustering. In order to obtain the region for each vertebra, distances between two consecutive vertebrae are calculated and their centroids are found. Then perpendicular separating lines are found on these centroids by applying affine transformation on the line passing through centroid and joining two vertebra. After rotation, these lines are combined to obtain 5 regions for each vertebra (C_3–C_7). The proposed method has been tested using 50 radiographs from which 250 vertebrae are detected. The dataset '$NHANESII$' used is publically available at National Library of Medicine (NLM). The results show the validity of presented technique.

Keywords: Vertebra segmentation · Region extraction · Affine transformation

1 Introduction

Vertebra segmentation and localization are the essential steps in many medical applications. For many spine surgeries correct vertebra segmentation with exact shape, localization is of great significance to make the treatment successful which is possible when correct regions of vertebrae are extracted. Wrong treatment may lead to more abnormalities. They help in the diagnosis of conditions like osteoporosis, spinal fractures and cervical trauma [18].

A. Abraham et al. (eds.), *Proceedings of the 16th International Conference on Hybrid Intelligent Systems (HIS 2016)*, Advances in Intelligent Systems and Computing 552,
DOI 10.1007/978-3-319-52941-7_4

Accurate detection and extraction of the spinal cord is a standard move for the segmentation of vertebrae which is a tiring task when performed manually. In literature, many segmentation techniques have been proposed such as 'Active Shape Model (ASM)' [11] and 'Generalized Hough Transform (GHT)' [4], etc. In most of the cases exact vertebral column extraction is the first step to be performed for accurate results. Benjelloun et al. [5] presented a methodology for segmentation of radiographs based on two algorithms. For the analysis and estimation of spine, a semiautomatic methodology was proposed. Larhmam et al. [15] presents a methodology based on template matching. For localization of vertebra they used Generalized Hough transform then K-means clustering is applied to get centroids of vertebra. The proposed methodology is tested using 66 x-ray images. Lecron et al. [16] developed a framework for x-ray images to localize the vertebra using ASM. This method reduced the computation time by synchronizing more than one CPUs and GPUs. Benjelloun et al. [6] developed a technique for the detection of vertebra boundaries using polar signature representation and segmentation of vertebra using watershed algorithm. In [7] they performed the identification and segmentation based on Active Shape Model. Larhman et al. [14] presented a technique based on offline training. A mean model of cervical vertebra is created and used as a template of GHT for vertebra localization. Then adaptive filter post processing is used to get the vertebra centroids securing the accuracy of 89% when tested on 40 x-ray images. Klinder et al. [13] proposed a framework to detect, identify and segment out the vertebra using CT scans. GHT is applied for the detection of vertebra then adapted triangulation shape is used for localization and segmentation of each vertebra individually. The presented method reported 70% identification accuracy. Casciaro et al. [10], obtained 83% accuracy for the localization of vertebra using local phase measure. Dong et al. [12] developed method to identify the visible vertebra using x-ray images. Mahmoudi et al. [17] presented a technique for cervical spine segmentation. The centroids obtained by single click were used to plot first order polynomial. Then using correlation, intervertebral locations were obtained and boundary lines are created. In each region formed by boundary lines contours are detected for each vertebrae to be segmented. The proposed methodology has given good accuracy using 100 testing images of real patients. Alomari et al. [19] designed a technique based on two stages for inter-vertebral discs. They performed experimental evaluation using two datasets containing 50 and 55 MRI cases. They obtained accuracy of 87% in case of 50 MRI and 89.1% with 55 MRI cases.

Our contribution in this regard is to develop an algorithm for the extraction of region for each vertebra in cervical spine X-rays. We will be using this region for vertebrae segmentation in future. Our proposed methodology of region extraction secure high accuracy of 96.88%. This method is based on shape based analysis, unsupervised clustering, and some basic functions are used to generate boundaries for each vertebra (C_3–C_7). This paper comprises of five sections. Section 2 includes the proposed methodology, Sect. 3 comprises the experimental evaluation and discussion is based on complete methodology. Section 4 includes the conclusion of proposed methodology.

2 Proposed Method

The presented technique is combination of shape based analysis, unsupervised clustering and Affine transformation [3] for the extraction of vertebrae regions. Figure 1 illustrates the presented technique with three main phases, blue block represents the 'Offline Training' phase, red block represents the 'Vertebra Detection' phase and the green block represents the main targeted phase 'Vertebra Regions Extraction'.

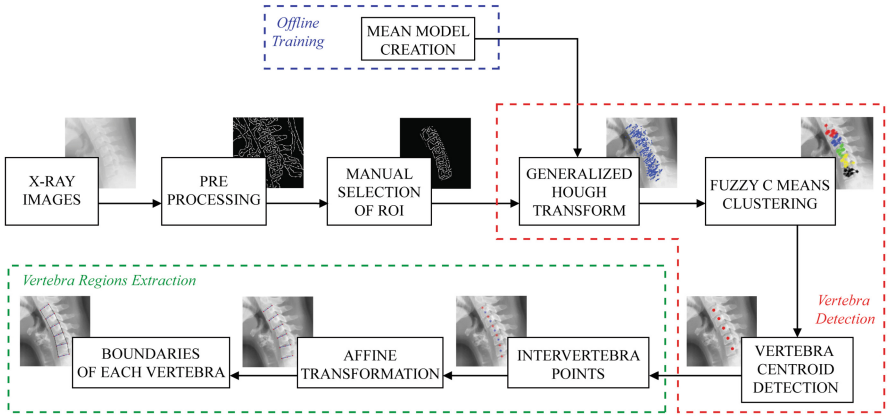


Fig. 1. Flow diagram of the proposed system, including its three main phases offline training, vertebrae detection and vertebra regions extraction

2.1 Offline Training

In offline training of the proposed methodology a mean model is created which is used as a template of GHT. This mean model is formed using a subset of 20 vertebrae of the same dataset. The contour is selected manually, used to generate mean model using Eq. (1) where 'N' represents the total no. of vertebra (N = 20) and 'V' represents the individual vertebra.

$$Mean\_Ver = \frac{1}{N}\sum_{i=0}^{N} V_i \tag{1}$$

2.2 Pre-processing

Contrast Enhancement. The low contrast of radiographs is one of the very basic reason for them to be challenging for image processing applications. So they need to be improved for any processing such as vertebra detection or extraction of vertebrae regions. For this purpose 'Contrast Limited Adaptive Histogram Equalization' (CLAHE) [1] is applied to the input radiographs, which enhance

the contrast by calculating local histogram of the specific region. It also minimizes the noise amplification. Then edges are detected using 'Canny Edge Detector' [9], as GHT works with edges only.

Region of Interest (ROI). The selection of ROI is carried out manually, covering the area of C_3 to C_7 vertebrae.

2.3 Vertebra Detection

After selection of ROI, GHT is applied on edges for the detection of cervical spine (C_3–C_7). The mean model created is used as a template of GHT. GHT works with two main divisions R-table and Accumulator. The R-table is basically the representation of template, constructed using position and direction of the edge points. Accumulator is basically a voting scheme, in which gradient is computed for each edge point and then voting is carried out for each location. Next, local maxima is computed for the detection of cervical spine (C_3–C_7).

After getting the voting results obtained from GHT, we applied Fuzzy C Mean (FCM) [8] algorithm to obtain the five clusters representing the centroids of cervical spine (C_3–C_7). In each iteration, membership is assigned to each data point and algorithm converges if two consecutive iterations give same result. Data point is assigned to the cluster with highest membership. Then centroid (C_i) for each cluster is obtained using data points and membership calculated.

2.4 Vertebrae Regions Extraction

After applying FCM we obtained five centroids representing five cervical spines. These five centroids are used further to obtain the inter-vertebrae location using Eq. (2), where C_i represents the centroid.

$$Inter\_C_i = (C_i + C_{i+1})/2 \quad where \quad i = 1, 2, 3, ..., C-1 \tag{2}$$

We get 4 inter-vertebra locations in the form of x and y co-ordinates but still two more points are required, one above the $C3$ and second after the $C7$ to obtain the region for each vertebra including $C3$ and $C7$. So we simply take the difference between $C1$ and first inter-vertebra location and subtract the obtained difference from $C1$ coordinates. This can be performed using Eq. (3)

$$P1 = C_1 - (Inter\_C_1 - C_1) \tag{3}$$

Similarly, second point is obtained by using Eq. (4) by taking the difference of C_5 centroid and inter-vertebra location of C_4 and C_5, then adding this difference in the coordinates of C_5.

$$P2 = C_5 + (C_5 - Inter\_C_4) \tag{4}$$

So, the final data we get for the Affine transformation are the 11 points in sequence as final data = P1, C_i, Inter_C_i, ..., Inter_C_i, $C_i + 1$, P2 where

$i = 1, 2, 3, 4$. After obtaining the inter-vertebra locations, angle Θ is computed using Eq. (5). Θ_1 and Θ_2 both are the angles at inter-vertebra location where Θ_1 is w.r.t previous centroid and Θ_2 is w.r.t next centroid. The final Θ is the mean angle of these two.

$$\Theta = \tan^{-1} \frac{C_{i,i+1}}{Inter\_C_i} \tag{5}$$

We used Affine transformation for rotating the centroids along the vertebra at its inter-vertebra location. Θ obtained help in this regard. We use 3 points, two inter-vertebra locations and one centroid to draw a line. Then this line is rotated using Affine transformation at angle $\Theta + 90°$, making it parallel to vertebra. Similarly, six lines are formed parallel to vertebra at inter-vertebra locations and covering the five cervical spines (C3–C7). End points of these lines are connected to extract the regions for each vertebra.

3 Results and Evaluation

Subjective evaluation is performed using dataset publically available at 'The National Health and Nutrition Examination Survey NHANES II' [2]. The mentioned dataset comprises 17000 x-ray images including 10000 cervical and 7000 lumbar spine images of various candidates under different conditions and orientation. Most of the papers are using subset of same dataset but no one has mentioned that on which basis they have selected these images. So we used 50 randomly selected images of NHANES II for testing and the second dataset is the set of 25 vertebrae created manually used for mean model creation. For validation of vertebrae detection, the subset of 50 images are visualized and center points are marked manually from C_3 to C_7. Figure 2 illustrates the validation of our proposed technique.

The difference between the centroids obtained as a result of FCM and the annotated centroids is obtained to get the distance between them. It shows the difference for each cervical spine for all the images of dataset. Red line represents the mean distance error obtained by the calculated difference. We can see that in few cases the distance is much more as compare to the other images which is due to the noise or other structures which makes the detection more challenging. Our main focus is to extract the regions for each vertebra (C_3–C_7), Fig. 3 shows the results obtained of few cases. First column shows input radiographs, second gives the voted vertebrae of GHT, third gives the centroids obtained from FCM, fourth shows the Affine transformation at inter-vertebra locations and last column gives the extracted regions for each vertebrae from C_3 to C_7. It is observed that in very few cases the regions obtained are not covering the vertebra because of there irregular shape. But we can deal with it by $\pm$ threshold in the co-ordinates of line and increasing its width.

In [17], vertebra segmentation is performed using region selection technique. They selection of region was carried out by the interaction of user where one has to click once at the center of vertebra body to initialize the processing. Then template matching was performed to obtained the intervertebral point

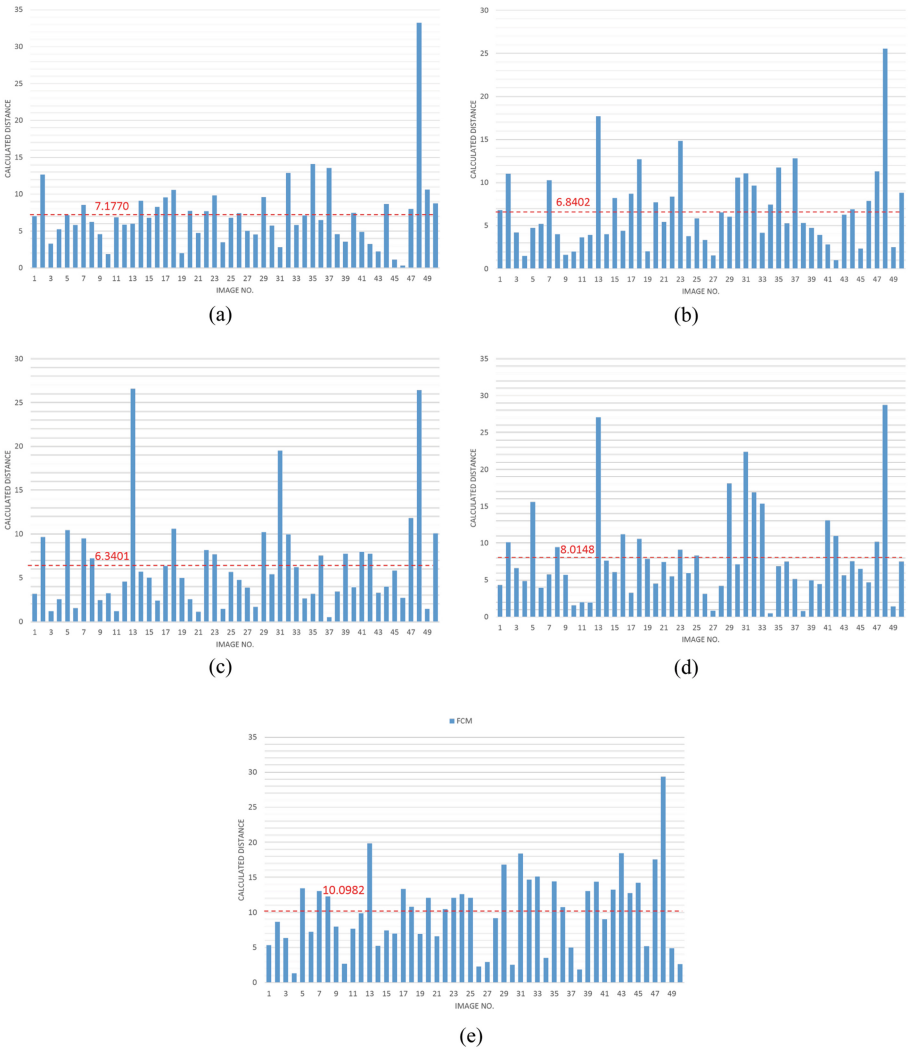


Fig. 2. Distance graph of annotated and calculated centroids for each vertebra with red doted line representing the mean distance error without threshold (a) distance graph of C3 (b) distance graph of C4 (c) distance graph of C5 (d) distance graph of C6 (e) distance graph of C7

and boundaries parallel to vertebra were obtained by using first order polynomial and then regions are extracted where as in our methodology a reference image of vertebra body is used to obtained the automated centroids of vertebra body using shape based analysis and clustering technique. Then using basic techniques intervertebral points are obtained. And finally Affine transformation form a rotated line parallel to the body of vertebra which extracted the region of each cervical spine.

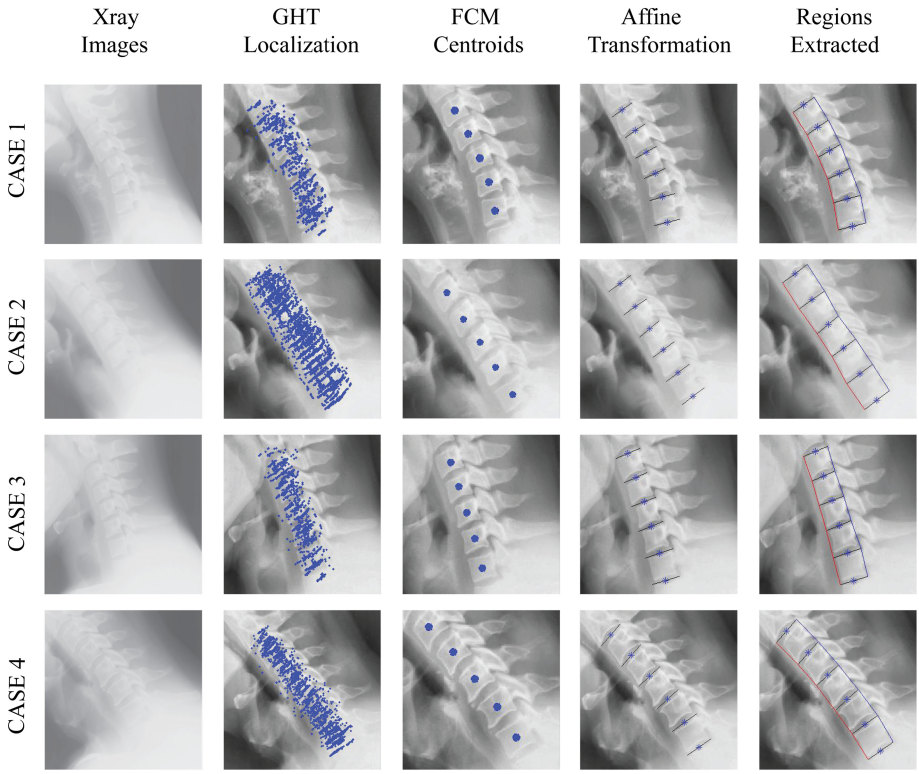


Fig. 3. Visual representation of results obtained of proposed technique for vertebrae regions extraction

The high accuracy of 96.88% is achieved when proposed methodology is tested using 250 vertebrae of 50 images. The contrast enhancement plays a vital role in the detection of vertebrae. Therefore, CLAHE is applied to enhance the contrast and improve the results.

4 Conclusion

Our presented technique is a novel methodology used to extract the regions of each cervical vertebra (C_3–C_7) which we will be using for segmentation of vertebrae, in near future. The subset of 50 cervical spine radiographs of NHANES II dataset has been used for testing of our technique. The proposed methodology is mainly based on GHT, FCM and Affine transformation for the extraction of areas covered by (C_3–C_7). The goal was to extract the regions for each vertebrae separately. So a mean model was created which we have used as a template in GHT and detect the vertebrae area, then FCM is applied and centroids are obtained. By using these centroids we computed the inter-vertebra locations and using Affine transformation, transform the centroids are transformed parallel to

the vertebra at this location. The proposed technique has shown satisfactory results with the accuracy of 96.88%. The future work is focused on the segmentation vertebrae (C_3–C_7) using these extracted regions.

References

1. Adaptive histogram equalization. https://en.wikipedia.org/wiki/Adaptive_histogram_equalization. Accessed 15 Jan 2016
2. NHANES II X-ray Images. https://ceb.nlm.nih.gov/proj/ftp/ftp.php. Accessed 27 Jan 2016
3. Affine transformation. https://en.wikipedia.org/wiki/Affine_transformation. Accessed 2 June 2016
4. Ballard, D.H.: Generalizing the hough transform to detect arbitrary shapes. Pattern Recogn. **13**(2), 183–194 (1991)
5. Benjelloun, M., Mahmoudi, S.: X-ray image segmentation for vertebral mobility analysis. Int. J. Comput. Assist. Radiol. Surg. **2**(6), 371–383 (2008)
6. Benjelloun, M., Mahmoudi, S.: Spine localization in x-ray images using interest point detection. J. Digital Imaging **22**(3), 309–318 (2009)
7. Benjelloun, M., Mahmoudi, S., Lecron, F.: A framework of vertebra segmentation using the active shape model-based approach. J. Biomed. Imaging **2011**, 9 (2011)
8. Bezdek, J.C.: Pattern Recognition with Fuzzy Objective Function Algorithms. Springer, Heidelberg (2013)
9. Canny, J.: A computational approach to edge detection. IEEE Transactions on Pattern Anal. Mach. Intell. **6**, 679–698 (1986)
10. Casciaro, S., Massoptier, L.: Automatic vertebral morphometry assessment. In: 2007 29th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, pp. 5571–5574. IEEE (2007)
11. Cootes, T.F., Taylor, C.J., et al.: Statistical models of appearance for computer vision (2004)
12. Dong, X., Zheng, G.: Automated vertebra identification from X-Ray images. In: Campilho, A., Kamel, M. (eds.) ICIAR 2010. LNCS, vol. 6112, pp. 1–9. Springer, Heidelberg (2010). doi:10.1007/978-3-642-13775-4_1
13. Klinder, T., Ostermann, J., Ehm, M., Franz, A., Kneser, R., Lorenz, C.: Automated model-based vertebra detection, identification, and segmentation in CT images. Med. Image Anal. **13**(3), 471–482 (2009)
14. Larhmam, M.A., Mahmoudi, S., Benjelloun, M.: Semi-automatic detection of cervical vertebrae in x-ray images using generalized hough transform. In: 2012 3rd International Conference on Image Processing Theory, Tools and Applications (IPTA), pp. 396–401, October 2012
15. Larhmam, M.A., Benjelloun, M., Mahmoudi, S.: Vertebra identification using template matching modelmp and k-means clustering. Int. J. Comput. Assist. Radiol. Surg. **9**(2), 177–187 (2014)
16. Lecron, F., Mahmoudi, S.A., Benjelloun, M., Mahmoudi, S., Manneback, P.: Heterogeneous computing for vertebra detection and segmentation in x-ray images. J. Biomed. Imaging **2011**, 5 (2011)
17. Mahmoudi, S., Benjelloun, M.: A new approach for cervical vertebrae segmentation. In: Rueda, L., Mery, D., Kittler, J. (eds.) CIARP 2007. LNCS, vol. 4756, pp. 753–762. Springer, Heidelberg (2007). doi:10.1007/978-3-540-76725-1_78

18. Nytimes: Spinal Cord Trauma. http://www.nytimes.com/health/guides/disease/spinal-cord-trauma/overview.html. Accessed 24 April 2016
19. Raja'S, A., Corso, J.J., Chaudhary, V.: Labeling of lumbar discs using both pixel- and object-level features with a two-level probabilistic model. IEEE Trans. Med. Imaging **30**(1), 1–10 (2011)

Automatic Extraction of Text and Non-text Information Directly from Compressed Document Images

Mohammed Javed[1(✉)], P. Nagabhushan[2], and Bidyut B. Chaudhuri[3]

[1] Department of Computer Science and Engineering, NMAM Institute of Technology (Affiliated to VTU, Belagavi), Nitte 574110, India
javedsolutions@gmail.com, javed@nitte.edu.in
[2] Department of Studies in Computer Science, University of Mysore, Mysuru 570006, India
[3] CVPR Unit, Indian Statistical Institute, Kolkata 700108, India

Abstract. Texts, images, audios, and videos form the major volume in Big Data being generated in today's tech-savvy world. Such data are preferably archived and transmitted in the compressed form to realize storage and transmission efficiency. Through compression, though data becomes storage and transmission efficient, its processing gets expensive as it requires decompression as many times the data needs to be processed; and this requires additional computing resources. Therefore it would be novel, if the data processing and information extraction could be carried out directly from the compressed data without subjecting it to decompression. In this backdrop, the research paper demonstrates a novel technique of extracting text and non-text information straight from compressed document images (supported by TIFF and PDF formats) using the correlation-entropy features that are directly computed from the compressed representation. The experimental results reported on compressed printed text document images validate the proposed method, and also demonstrate the fact that the text and non-text information extracted from the compressed document are identical to that obtained from uncompressed representation.

Keywords: Compressed document processing · Run-length compressed domain · Text and non-text · Correlation-entropy

1 Introduction

With the rapid growth of digital libraries, e-governance, and internet based applications huge amounts of data (texts, images, audios and videos) are being generated and archived in the compressed form with the intention of providing better storage and transfer efficiencies. No doubt due to compression, data storage and transmission become efficient, but unfortunately its processing becomes expensive as it requires decompression (to operate) and re-compression (again to make

A. Abraham et al. (eds.), *Proceedings of the 16th International Conference on Hybrid Intelligent Systems (HIS 2016)*, Advances in Intelligent Systems and Computing 552,
DOI 10.1007/978-3-319-52941-7_5

the data suitable for archival and transmission) which are expensive operations, and moreover they need the operation to be executed as many times the data needs to be processed. This limitation induces motivation to look for a novel technology that can process and extract the required information directly from the compressed data without undergoing the stage of decompression. Demonstrating this novel idea for extracting text and non-text information directly from compressed document images is the prime objective of this research paper.

Text is an important source of information in documents related to journals, patents, magazines, newspapers, books, etc. Along with text it is also very common to include non-text components like images, tables and graphs as additional information. However in today's scenario, with the availability of modern editing and publishing tools like CorelDRAW, EditPlus, CKEditor which provide the flexibility of embedding non-text components in various complex layouts, the task of automatic extraction of text and non-text information has become more challenging. Specifically from the Document Image Analysis (DIA) perspective, the knowledge of text and non-text information is very essential in Optical Character Recognition (OCR) and many document related applications [12,13] like postal automation, bank cheque processing, form processing, and so on. As a consequence, in the literature [2,3,12–14], many text and non-text segmentation techniques have been proposed to detect and extract text and non-text information from document images. However, the available off-the-shelf techniques are specifically tuned to work with uncompressed documents, and in case of a compressed document it has to be accomplished through decompression. Therefore, in the current research paper, a novel technique of processing (compressed data) and extracting text and non-text information directly from compressed printed document images without involving the stage of decompression is underscored. The rest of the paper is organized as follows: Sect. 2 introduces the proposed model and outlines the procedure of extracting text and non-text information from compressed document images, Sect. 3 reports experimental results, and Sect. 4 summarizes the research paper.

2 Proposed Method

The proposed model for extracting text and non-text information directly from compressed printed document images is shown in Fig. 1. In the proposed model, document images compressed using TIFF supported compression schemes like CCITT Group 3 (1D and 2D) and CCITT Group 4 2D which represent different flavors of run-length data are taken as input. The run-length data is generated using the Run-Length Compressed Domain (RLCD) model proposed in the works of [4,5,7,8,11], and the same model is used here for extracting text and non-text information. Suppose a binary image ($m \times n$ pixels) consisting of m rows and n columns is compressed, then the run-length compressed matrix generated will be of the size ($m \times n'$ runs) where $n' < n$. For example if the binary image contains a row with 00110001110001 pixels, then the compressed row will have 223331 runs. Incidently this type of compressed data represents alternating

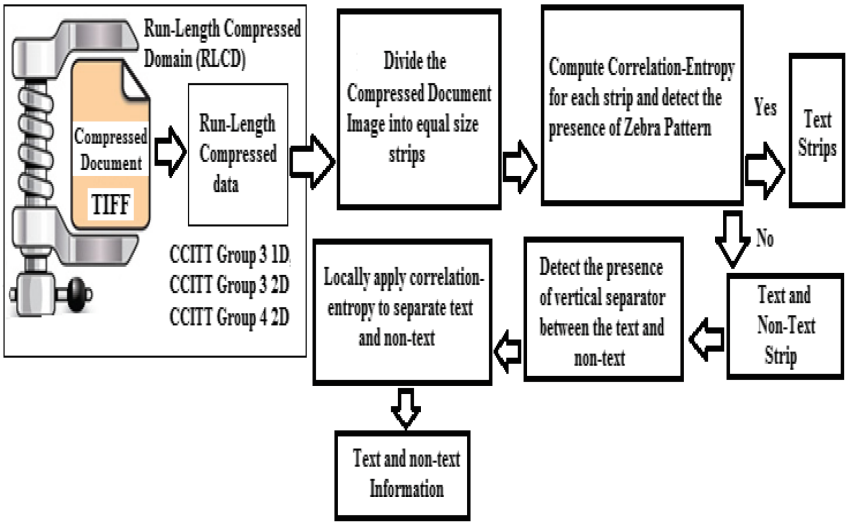


Fig. 1. Proposed model

white and black runs throughout the row, and in the compressed run-matrix they are positioned respectively at odd and even columns (as per the CCITT Standard).

The first step in extracting text and non-text information from the compressed document image is to divide the document into equal size strips (see Fig. 2), and then use each compressed strip independently to compute the correlation-entropy feature proposed in [9]. The strip division shown in the uncompressed document (Fig. 2) are rectangular in shape, but in the compressed data the strips may not be rectangular, and the compressed data from the respective strips needs to be extracted by keeping track of the cumulative runs and residue runs at the respective strip positions (refer paper [6]). The strip division in case of an uncompressed binary image and its equivalent compressed run-length matrix are shown in Fig. 3(a) and (b). The compressed data extracted for the strip number 2 is shown in Fig. 3(c).

In a typical binary (0 and 1) image, the number of possible transitions between any two consecutive rows could be $0-0$, $1-1$, $0-1$, and $1-0$. In the work presented in [9], the same transition values are referred to as correlation (D_{0-0} and D_{1-1}) and entropy (D_{0-1} and D_{1-0}) features. The researchers have used the North West Corner method to compute all the four features from the compressed document image. In order to give a clear picture of computing the correlation-entropy feature, the North West Corner method is illustrated with an example taking two rows of compressed runs $R_A(i) = [12122231]$ and $R_B(j) = [22112222]$) vertically and horizontally in Table 1. The feature computed in Table 1 is decided by the index values of i and j shown in Table 2.

Fig. 2. A sample document showing the logical division in the form of equal size strips and the presence of text and non-text information (yellow rectangle) in the uncompressed version

However, in the current paper as the prime focus is on extracting text and non-text information based on *zebra patterns*, it is sufficient to compute the correlation D_{0-0} feature from the compressed document. Therefore, the method presented in [9] is modified to extract only the correlation feature focusing on odd columns of the run-length matrix. The correlation-entropy feature is basically computed by taking every two consecutive rows (containing runs) of the compressed document. For example, consider first two consecutive rows in the compressed data of Fig. 3(b) with first row (162266261) as the source runs and the second row (16226621411) as the destination runs. The problem now is to map all the corresponding runs with $0-0$ pixel transitions from the source to destination and count the number of such mappings which indicates the D_{0-0} feature which is 12 (see Fig. 3(d)). In the next step, the previous destination row

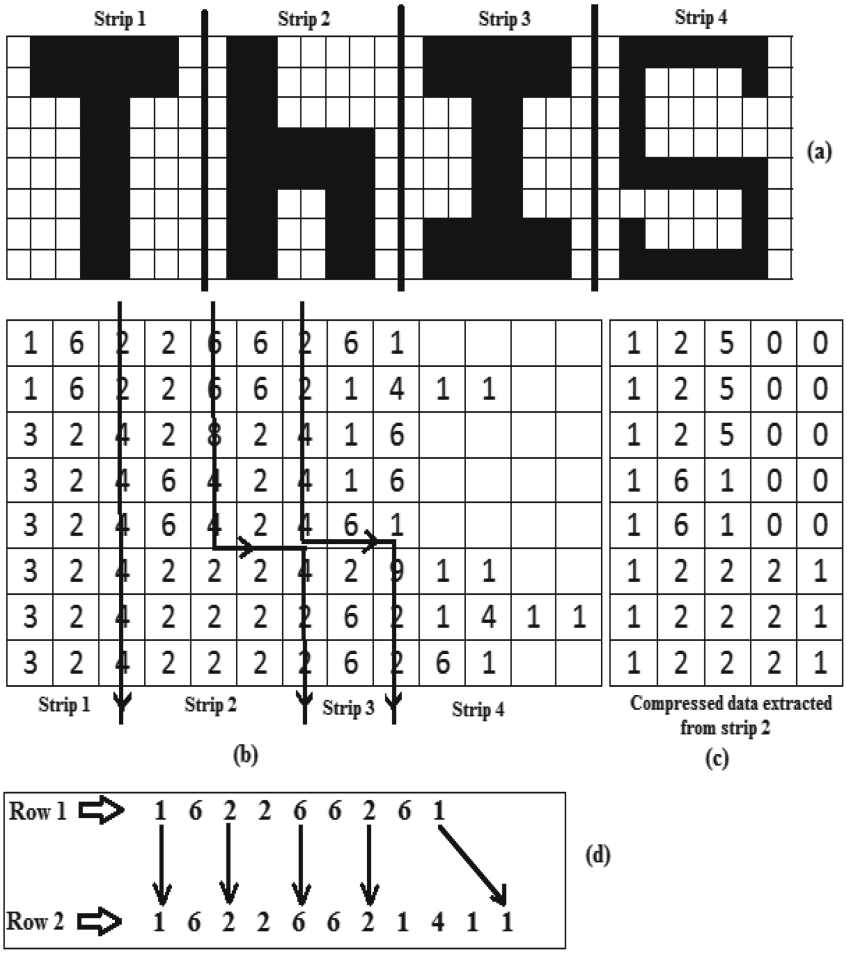


Fig. 3. Illustrating the division of sample binary image into 4 strips (a) in the uncompressed version (b) in the run-length compressed version (c) compressed data extracted for the strip number 2 (d) modified correlation computation

becomes source row and the next row in the matrix becomes destination row, and the whole process is repeated. Similarly the D_{0-0} feature is computed for the entire document. In the D_{0-0} feature the presence of the informative pixel is indicated by some non-zero value and its absence is indicated by a zero value. We represent all non-zero values as high (1) value and all zero values as low (0) value. In case of a text block, this representation produces an alternating high and low value regions (see Fig. 4(a)) is denoted as *zebra patterns*. In the text block the high regions refer to text lines and low regions refer to space between the text lines. In case of non-text block (images, table, and graphs) an irregular pattern which does not resemble the zebra pattern will be observed.

Table 1. Computing correlation-entropy from the run vectors $R_A(i)$ and $R_B(j)$ (where i and j indicate the position of runs in run vectors)

$i_{\downarrow}/j_{\rightarrow}$	2	2	1	1	2	2	2	2	i	j	D_{0-0}	D_{1-1}	D_{0-1}	D_{1-0}
1	1								{1}	{1}	1			
2	1	1							{2}	{1,2}		1		1
1		1							{3}	{2}			1	
2			1	1					{4}	{3,4}		1		1
2					2				{5}	{5}	2			
2						2			{6}	{6}		2		
3							2	1	{7}	{7,8}	2		1	
1								1	{8}	{8}		1		
											5	5	2	2

Table 2. Strategy employed to compute correlation-entropy feature by mapping $i \rightarrow j$ from the run vectors $R_A(i)$ and $R_B(j)$

Case	$R_A(i)$	$R_B(j)$	Feature
1	Odd(i)	Odd(j)	D_{0-0}
2	Even(i)	Even(j)	D_{1-1}
3	Odd(i)	Even(j)	D_{0-1}
4	Even(i)	Odd(j)	D_{1-0}

With the correlation feature (D_{0-0}), the strip containing text information forms a zebra pattern (see Fig. 4(a)), whereas the strip containing non-text information do not form a zebra pattern (see Fig. 4(b)). Further, once the strip containing non-text information is detected (see Fig. 4(b)), the strip region containing the non-text information is locally analyzed to find out the vertical separator between the text and non-text. This is because, in a document with Manhattan layout, the text and non-text generally have vertical separators [10]. If any vertical separator is detected (see Fig. 5(a) and (b)), then both the text and non-text information on its either side are further analyzed for the presence of zebra pattern. The non-text region which does not produce a zebra pattern is identified as non-text information in the compressed document image (see Fig. 2 shown within yellow rectangles). Apart from images, the absence of the zebra pattern is also observed with non-text regions like graphs and tables (with line separators) and hence the proposed method also works successfully for such non-text regions.

3 Experimental Results

The proposed method was experimented with a dataset of 50 compressed document images (free from noise and skew) collected from the Prima Layout Dataset

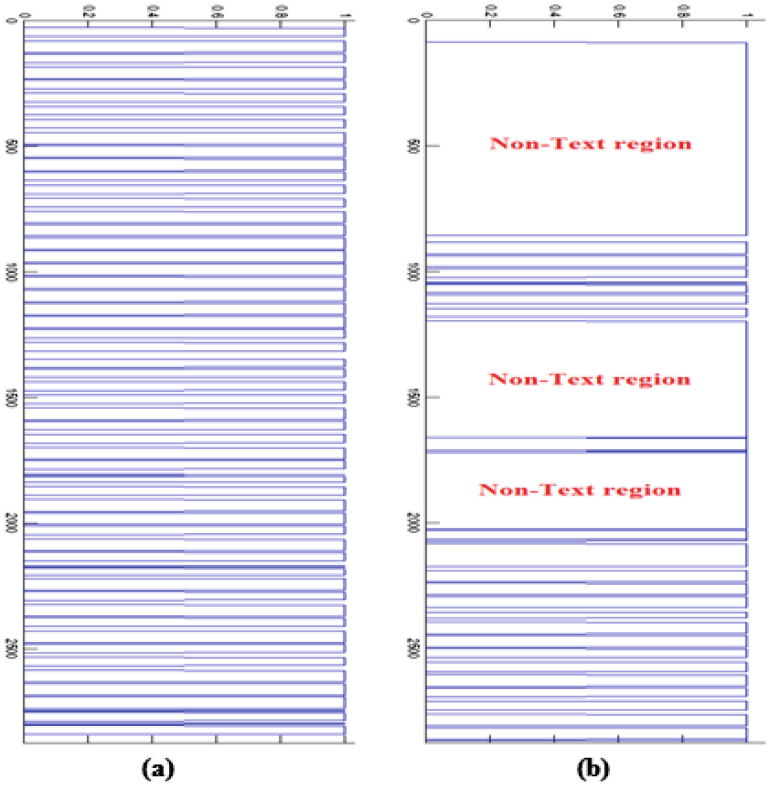


Fig. 4. Zebra pattern in case of a strip containing (a) only text (strip number 1) and (b) text and non-text (strip number 3)

[1] and also from text books and student project reports. Each document consists of text regions, and non-text regions like images, tables (with line separators), and graphs (within rectangular enclosure). The documents are of single, double and three column Manhattan layouts. The detected text and non-text regions that share 90% of pixel similarity with respect to ground truth were taken as correctly extracted regions. The accuracy of extracting text and non-text information directly from compressed document image dataset was reported to be in the range 91%–97% and 82%–94% respectively. Further, through the experimental results it was also observed that the text and non-text information extracted from the compressed document were identical to that of uncompressed version (see Table 3). The observation made is supported by the fact that since run-length is lossless compression the features extracted and the analysis carried out in the compressed representation should be identical to that of uncompressed representation.

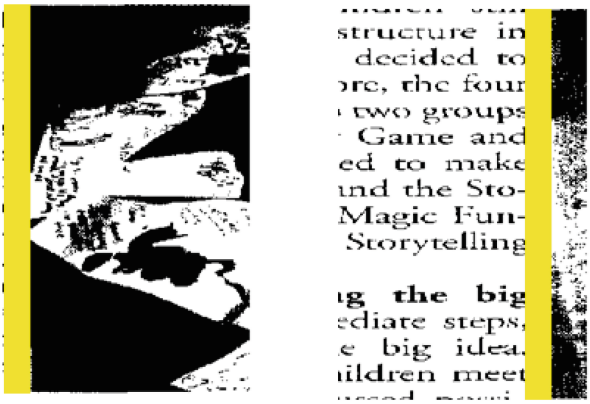


Fig. 5. Local analysis of the non-text regions detected through zebra pattern (a) strip containing non-text boundary (b) strip containing text and non-text boundary

Table 3. Accuracy of text and non-text information extracted from compressed and uncompressed versions show identical results

	Document	Accuracy with text	Accuracy with non-text
1	Compressed version	91%–97%	82%–94%
2	Uncompressed version	91%–97%	82%–94%

4 Conclusion

In this paper, the novel idea of processing and extracting information directly from compressed data was demonstrated taking the case study of extracting text and non-text information from compressed document images. The proposed method partitioned the input compressed document image into equal size strips, and then the correlation-entropy based feature was employed for extracting text and non-text information using the concept of the zebra pattern. The presence of the zebra pattern indicated text and the absence of the zebra pattern was used as clue for locating the non-text information. The strip containing both text and non-text components were further analyzed locally with the zebra patterns and the non-text components were removed. The proposed method was experimented with compressed printed text document images and the results reported validate the proposed idea.

References

1. Antonacopoulos, A., Bridson, D., Papadopoulos, C., Pletschacher, S.: A realistic dataset for performance evaluation of document layout analysis. In: Proceedings of the 10th International Conference on Document Analysis and Recognition, (ICDAR2009), Barcelona, Spain, pp. 296–300 (2009)

2. Breuel, T.M.: High performance document layout analysis. In: Proceedings of Symposium on Document Image Understanding Technology, April 2003
3. Chen, K., Yin, F., Liu, C.L.: Page segmentation with efficient whitespace rectangles extraction and grouping. In: 12th International Conference on Document Analysis and Recognition, pp. 958–962 (2013)
4. Javed, M.: On the possibility of processing document images in compressed domain. Ph.D. thesis, Department of Studies in Computer Science, University of Mysore (2016)
5. Javed, M., Krishnanand, S.H., Nagabhushan, P., Chaudhuri, B.B.: Visualizing ccitt group 3 and group 4 tiff documents and transforming to run-length compressed format enabling direct processing in compressed domain. Procedia Comput. Sci. **85**, 213–221 (2016). (International Conference on Computational Modelling and Security - CMS 2016)
6. Javed, M., Nagabhushan, P., Chaudhuri, B.B.: Direct processing of run-length compressed document image for segmentation and characterization of a specified block. Int. J. Comput. Appl. (IJCA) **83**(15), 1–6 (2013)
7. Javed, M., Nagabhushan, P., Chaudhuri, B.B.: Extraction of line-word-character segments directly from run-length compressed printed text-documents. In: National Conference on Computer Vision, Pattern Recognition, Image Processing and Graphics (NCVPRIPG), pp. 1–4 (2013)
8. Javed, M., Nagabhushan, P., Chaudhuri, B.B.: Extraction of projection profile, run-histogram and entropy features straight from run-length compressed documents. In: 2nd IAPR Asian Conference on Pattern Recognition (ACPR), pp. 813–817 (2013)
9. Javed, M., Nagabhushan, P., Chaudhuri, B.B.: Automatic extraction of correlation-entropy features for text document analysis directly in run-length compressed domain. In: 13th International Conference on Document Analysis and Recognition (ICDAR), pp. 1–5 (2015)
10. Javed, M., Nagabhushan, P., Chaudhuri, B.B.: Automatic page segmentation without decompressing the run-length compressed printed text documents. International Journal of Information Processing Systems (JIPS) (Accepted for Publication) (2015)
11. Javed, M., Nagabhushan, P., Chaudhuri, B.B.: A direct approach for word and character segmentation in run-length compressed documents and its application to word spotting. In: 13th International Conference on Document Analysis and Recognition (ICDAR), pp. 216–220 (2015)
12. Kasturi, R., Gorman, L.O., Govindaraju, V.: Document image analysis: a primer. Sadhana Part 1 **1**(27), 3–22 (2002)
13. Marinai, S.: Introduction to document analysis and recognition. Stud. Comput. Intell. (SCI) **90**, 1–20 (2008)
14. Zirari, F., Ennaji, A., Nicolas, S., Mammass, D.: A document image segmentation system using analysis of connected components. In: 12th International Conference on Document Analysis and Recognition, pp. 753–757 (2013)

Fuzzy Logic Dynamic Parameter Adaptation in the Gravitational Search Algorithm

Frumen Olivas, Fevrier Valdez, and Oscar Castillo(✉)

Tijuana Institute of Technology, Tijuana, Mexico
frumen@msn.com, {fevrier,ocastillo}@tectijuana.mx

Abstract. In this paper a new approach for parameter adaptation is proposed, where a fuzzy system is implemented to dynamically change some parameters of the Gravitational Search Algorithm (GSA), the idea of dynamically changing the parameters of GSA come from the necessity of having a method that allows GSA to be implemented on any problem without the need to find the best values for each parameter, because the fuzzy system will do that for us. To properly adjust the parameters the fuzzy system depends on some metrics of GSA, like the percentage of iterations elapsed and the degree of dispersion of the agents from GSA on the search space, which are used in the proposed approach.

Keywords: Fuzzy logic · Gravitational search algorithm · Dynamic parameter adaptation · GSA

1 Introduction

In the literature is well known that it is a critical issue to find the best parameters of an optimization method for a specific task [2], and this problem can be solved by the implementation of a methodology that changes these parameters automatically [6], like by using mathematical equations, fuzzy logic or even other optimization methods. The proposed methodology consists in performing several experiments to understand the behavior of GSA by changing their parameters and finding out what parameters have more influence in controlling the abilities of GSA to perform a local or a global search. Based on this a set of fuzzy rules can be designed in order to control these abilities of GSA, allowing to improve the search of GSA and to improve the quality of the results. The experimental results shows that our proposed methodology helps in the improvement of the quality of the results, and GSA now can be applied to a wide variety of problems without the need to change their parameters manually, however we will continue with the improvement of GSA through other forms or types of fuzzy logic. A comparison against the original GSA, other GSA improvements and our proposal is presented, where the results are promising, in this case considering a set of 15 benchmark mathematical functions. The main contribution of this paper is the analysis of the parameters of GSA to know the behavior and using this knowledge to control its abilities to perform a global or local search, also the use of a fuzzy system in the GSA to dynamically change some parameters, which helps in the design of the desired behavior of GSA. The GSA method has been applied to several applications: to edge detection in [13], to clustering problems in [4], for feature selection in [10],

A. Abraham et al. (eds.), *Proceedings of the 16th International Conference on Hybrid Intelligent Systems (HIS 2016)*, Advances in Intelligent Systems and Computing 552,
DOI 10.1007/978-3-319-52941-7_6

and approaches hybridized with PSO for function optimization in [7], and converted as multi-objective optimization problem in [3], as a prototype classifier in [1]. Unlike other methods where the *kbest* parameter has no change [8], we have a fuzzy system that dynamically change both the α and *kbest* parameters along the iterations.

2 Theoretical Basis

Fuzzy logic system helps in modeling complex problems, like the parameter adaptation in GSA. Fuzzy logic and fuzzy set theory were proposed by Zadeh in [14–16] brings the accessibility to design a fuzzy system with its membership functions and fuzzy rules. In our methodology the membership functions and rules were designed by knowledge or by trial and error. The membership functions represent the levels of the parameters to be controlled and the metrics of GSA, the fuzzy rules were created to design the desired behavior of the GSA, but also to control the abilities of GSA to perform a local or a global search. Rashedi originally proposed the gravitational search algorithm in [11], which is based on the law of gravity and second motion law [5]. In this case, every agent is evaluated and its mass depends on their fitness, the better the fitness the bigger or heavier the agent, all these agents are attracted each other by their gravity force, and causes a global movement of all agents, each mass is a solution and the algorithm is navigated by properly adjusting the gravitational and inertia masses. The agents are represented by

$$X_i = (x_i^1, \ldots, x_i^d, \ldots, x_i^n) \text{ for } i = 1, 2, \ldots, N, \tag{1}$$

where x_i^d represents the position of an agent i in the dimension d. To calculate the gravitational force acting on mass i from mass j in a specific time t, we have the following expression

$$F_{ij}^d(t) = G(t)\frac{M_{pi}(t) \times M_{aj}(t)}{R_{ij}(t) + \varepsilon}(x_j^d(t) - x_i^d(t)) \tag{2}$$

where M_{aj} is the mass related to agent j, M_{pi} is the mass related to agent i, $G(t)$ is gravitational constant at time t, ε is a small constant, and $R_{ij}(t)$ is the Euclidian distance between the two agents i and j. The acceleration is calculated as follows:

$$a_i^d(t) = \frac{F_i^d(t)}{M_{ii}(t)} \tag{3}$$

Where M_{ii} is the mass of the agent i and F_i is the gravitational force of the agent i. The velocity of an agent is defined with Eq. 4 where the new velocity is given by a fraction of the current velocity plus the new acceleration.

$$v_i^d(t+1) = rand_i \times v_i^d(t) + a_i^d(t) \tag{4}$$

The position of an agent is defined by Eq. 5, where the new position is given by the current position plus the new velocity given by Eq. 4.

$$x_i^d(t+1) = x_i^d(t) + v_i^d(t+1) \tag{5}$$

In GSA the gravitational constant G is defined by Eq. 6:

$$G(t) = G_0 e^{-\alpha t/T} \tag{6}$$

Where α is a user-defined value, t is the actual iteration and T is the maximum number of iterations. The way that GSA protects its better agents, this is the mechanism of elitism, which is by only allowing the agents with bigger mass apply their gravitational force to the other agents. In the beginning all the agents can apply their mass to all of the other agents and this is decreasing with time and at the end only a few agents or only the *kbest* agents will apply their gravitational force

$$F_i^d(t) = \sum_{j \in Kbest, j \neq 1} rand_i F_{ij}^d(t) \tag{7}$$

Where *Kbest* is a set of *kbest* agents with the best fitness values and the biggest masses, please distinguish between lowercase and uppercase, *kbest* is the number of agents in the *Kbest* set. Through an analysis of the equations of GSA, and performing several experiments changing the values of the parameters, the behavior of GSA can be defined and this opens the possibility of the parameter adaptation through an intelligent methodology, like fuzzy logic.

3 Methodology

The methodology proposed consists first in performing an analysis of the parameters in the equations of GSA, then perform experiments to understand the behavior of GSA, while its parameters are changed, in order to select the parameters that have the most impact on the behavior of GSA, also define the metrics of GSA which can help to control the behavior of GSA. Once the parameters and the metrics are defined then the fuzzy rules can be defined in order to control the abilities of GSA to perform a global or a local search, and based on the metrics the fuzzy system can change the GSA parameters and control its behavior. In the case of GSA, and after the analysis of the parameters on the equations and performing several experiments, the parameters chosen to be dynamically adjusted are α from Eq. 6 and *kbest* from Eq. 7. These parameters have the most impact on the behavior of GSA, allowing controlling the abilities of GSA to perform a global or a local search. The general scheme of the proposed methodology for dynamic parameter adaptation on GSA is illustrated in Fig. 1 where a fuzzy system is used to dynamically adjust the values of some parameters of GSA based on some metrics of GSA.

The α parameter was chosen because it has a big impact in the behavior of GSA, because it directly affects the gravitational constant G, therefore indirectly affects the

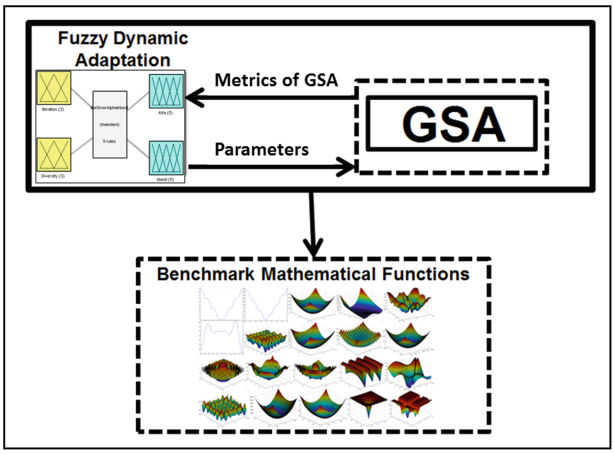


Fig. 1. Proposal for dynamic parameter adaptation using fuzzy logic

the gravitational force, and the velocity of the agents; with the control of the α parameter the velocity of the agents can be controlled, so it can be set to perform a global search if the velocity is high, or perform a local search if the velocity is low. The *kbest* parameter was chosen because it affects how many agents apply their gravitational force to the other agents; if all agents apply their gravitational force to all of the other agents, then all agents will move with no destination, but if only the best agent apply its gravitational force then all of the other agents will move towards the best agent and for this reason move to the best region of the search space.

The main metrics of GSA that we use are the number of iterations and diversity, for this purpose iteration is defined as the percentage of elapsed iterations given by Eq. 8, and the diversity is defined as the level of dispersion of the agents given by Eq. 9. The reason we use these metrics is because with the percentage of elapsed iterations we can decide if GSA has to perform a global or a local search (i.e. in early iterations we want that GSA performs a global search), and with the level of dispersion of the agents we ensure that the agents are separated from each other or together (i.e. if the agents are close together in early iterations we want that the agents search by their own).

$$Iteration = \frac{Current\ Iteration}{Maximum\ of\ Iterations} \tag{8}$$

where in Eq. 8, *current iteration number* is the number of elapsed iterations, and *maximum number of iterations* is the number iterations established for GSA to find the best solution.

$$Diversity(S(t)) = \frac{1}{n_s}\sum_{i=1}^{n_s}\sqrt{\sum_{j=1}^{n_x}\left(\chi_{ij}(t) - \bar{\chi}_j(t)\right)^2} \tag{9}$$

In Eq. 9, S is the population of the GSA; t is the current iteration or time, n_s is the size of the population, i is the number of agents, n_x is the total number of dimensions, j is the number of the dimension, x_{ij} is the j dimension of the agent i, $\bar{x}_j$ is the j dimension of the current best agent of the population. In the fuzzy system, the metrics of GSA are considered as inputs, and the output is *kbest*. Figure 2 shows the input variable iteration, with a granularity of three triangular membership functions, with a range from 0 to 1. The diversity as input variable with a range from 0 to 1 has been granulated into three triangular membership functions, and is represented in Fig. 3.

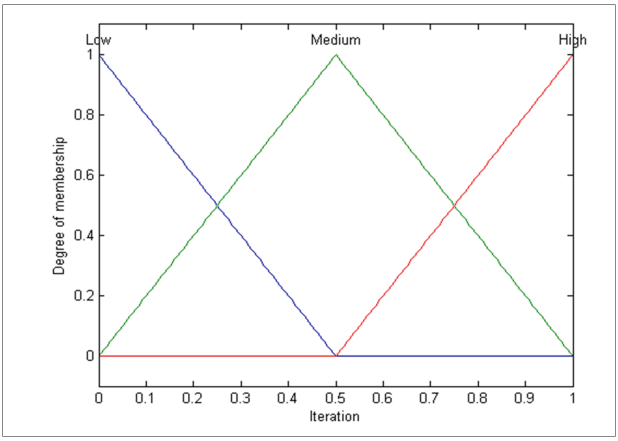


Fig. 2. Iteration as input variable for the fuzzy system

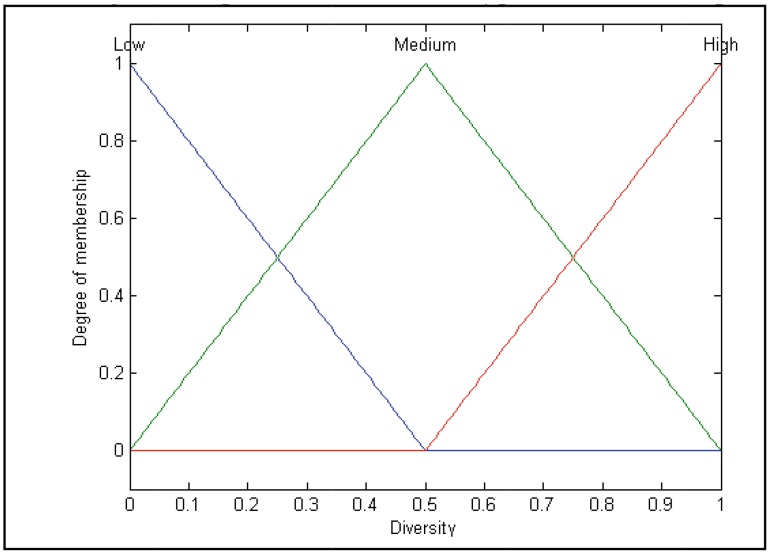


Fig. 3. Diversity as input variable for the fuzzy system

The iteration as input variable has a range from 0 to 1 because from Eq. 8, this given as a percentage that only can be from 0% to 100% in other words from 0 to 1. The granularity into three triangular membership functions is because with the increase in the number of membership functions in the inputs can cause that the number of possible rules increases exponentially, and with three membership functions we considered enough. The triangular type of membership functions was chosen based on an investigation reported on [9], which says that in this kind of problem the type of membership function doesn't make a big difference, in addition the membership functions is easier to configure compared with the other types of membership functions.

Diversity as input variable has a range from 0 to 1, because the results from Eq. 9 are between 0 and 1, and can be interpreted as if the population is close together, this means there is no diversity represented by 0, any other result means that there is diversity and with the result of 1 or more means that there is enough diversity. The α output variable has a range from 0 to 100 and has been granulated into five triangular membership functions represented in Fig. 4.

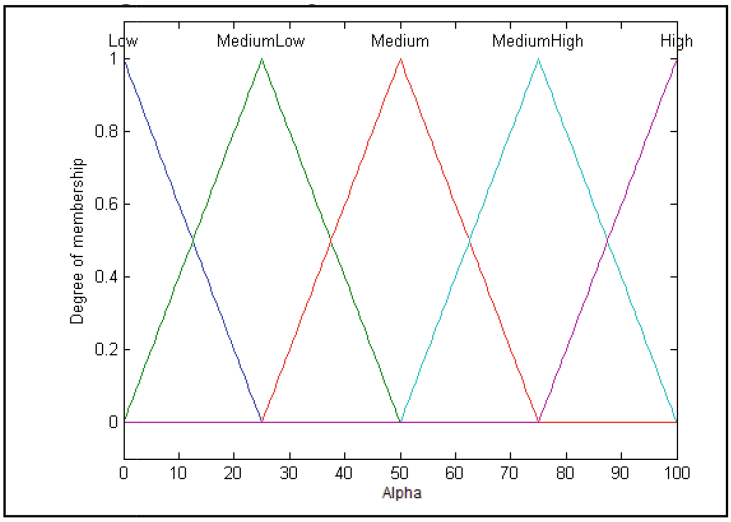


Fig. 4. Alpha "α" as output variable for the fuzzy system

The output variable kbest has a range from 0 to 1 and has been granulated into five triangular membership functions, and is shown in Fig. 5. The kbest parameter as output has been defined with a range from 0 to 1, because this means the percentage of the best agents that can apply their gravity force to all other agents, also has been granulated into five triangular membership functions, and this is because with the two input variables with three membership functions each resulting into nine possible rules when combined all with the "and" operator, with these many rules we need to granulated more to allow a better performance of the fuzzy system. The rule set used in this system is illustrated in Fig. 6 and there are a total of nine rules.

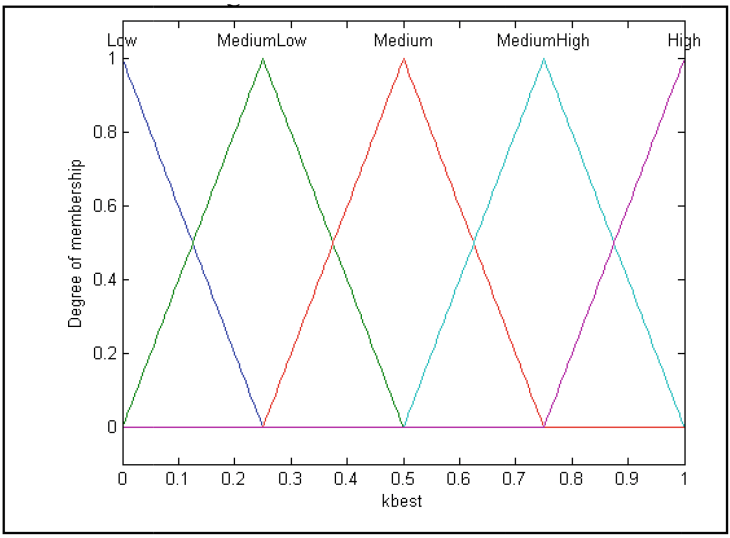


Fig. 5. kbest as output variable for the fuzzy system

1. If (Iteration is Low) and (Diversity is Low) then (Alfa is Medium)(kbest is High)
2. If (Iteration is Low) and (Diversity is Medium) then (Alfa is MediumLow)(kbest is MediumHigh)
3. If (Iteration is Low) and (Diversity is High) then (Alfa is Low)(kbest is Medium)
4. If (Iteration is Medium) and (Diversity is Low) then (Alfa is MediumHigh)(kbest is MediumHigh)
5. If (Iteration is Medium) and (Diversity is Medium) then (Alfa is Medium)(kbest is Medium)
6. If (Iteration is Medium) and (Diversity is High) then (Alfa is MediumLow)(kbest is MediumLow)
7. If (Iteration is High) and (Diversity is Low) then (Alfa is High)(kbest is Medium)
8. If (Iteration is High) and (Diversity is Medium) then (Alfa is MediumHigh)(kbest is MediumLow)
9. If (Iteration is High) and (Diversity is High) then (Alfa is Medium)(kbest is Low)

Fig. 6. Rule set used for the fuzzy system

The rule set was created based on some principles about the desired behavior for GSA, for example: in early iterations the GSA must use exploration, in final iterations GSA must use exploitation, and with a low diversity GSA must use exploration and with a high diversity GSA must use exploitation. In other words, when the GSA starts the iterations must perform a global search to find the best regions of the search space, and in final iterations GSA must perform a local search in the best region found so far. With the diversity is all about trying to avoid local minimum because if the diversity is low this means that all the agents are in a very small area, and if it is not in final iterations these agents must separate. The complete fuzzy system is shown in Fig. 6, where it can be noted the inputs and outputs, is of Mamdani type (the membership functions of the outputs are fuzzy sets) and has the number of fuzzy rules used. In order to create the fuzzy system illustrated in Fig. 7, a series of various fuzzy system designs were initially created with only one output and this was done to know the appropriate rule set needed to control a specific parameter. In other words, the rules to control α were created separately from the rules to control *kbest*, but then joined to control both parameters at the same time in the fuzzy system from Fig. 7.

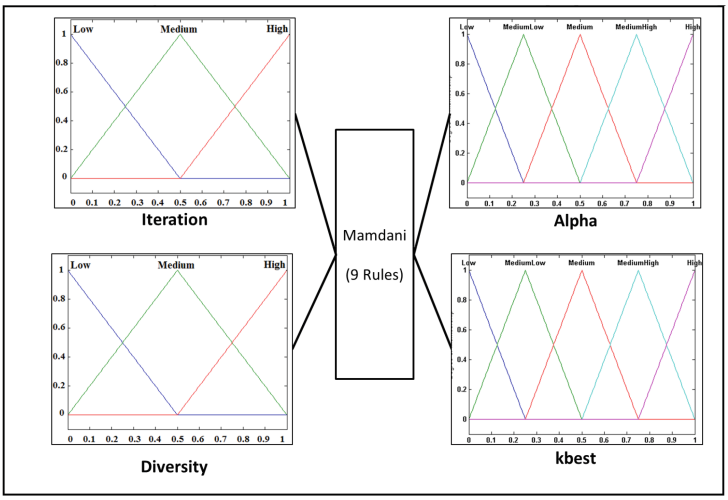


Fig. 7. Fuzzy system used to control the parameters of GSA

4 Experiments and Results

In the design of the experiments the GSA was applied to fifteen mathematical functions with the same constrains as the ones used in [12], and also to perform a comparison between the original GSA method from Rashedi et al. [11], the proposed approach, and a GSA with parameter adaptation from [12]. A total of 30 experiments for each function on the same conditions as the original proposal, with a maximum number of

Table 1. Comparison GSA [11] vs. our proposal

Function	GSA from [11]	Proposal
F1	7.3e-11	**5.35e-40**
F2	4.03e-5	**2.76e-19**
F3	0.16e-3	**7.61e-6**
F4	3.7e-6	**6.58e-11**
F5	25.16	**9.56**
F6	8.3e-11	**0**
F7	0.018	**6.15e-4**
F8	-2.8e + 3	**-2.95e + 3**
F9	15.32	**6.87**
F10	6.9e-6	**9.06e-25**
F11	0.29	**4.82e-5**
F12	0.01	**2.54e-4**
F13	3.2e-32	**3.45e-43**
F14	3.70	**1.27e-7**
F15	8.0e-3	**4.66e-8**

agents = 50, *maximum of iterations* = 1000, *dimensions* = 30, G_0 = 100 and α = 20, *kbest* is linear decreasing from 100% to 2%, however in our proposal α and *kbest* are dynamic, that is the only difference. In Table 1 we show the results of each method with each membership function, please note that each of the results in Table 1 is the average of 30 experiments, and the results in bold are the best when comparing our proposal with the original GSA from [11].

From the results in Table 1, we can notice that all the results in bold are in the proposal column and this means that the proposed approach always obtains the best results and outperform the results when compared with the original GSA method. And with the use of iteration and diversity as inputs the fuzzy system using the rule set provided we can control the behavior of α and kbest and thereby control the behavior of GSA. The comparison against other methods, in this case from Sombra et al. [12] in which they used a fuzzy system to only control α, and Particle Swarm Optimization (PSO) from Rashedi et al. [11]. Table 2 shows the results of these comparisons between our proposal and the other methods, and these results are obtained using the same conditions as in the Table 1, so each result is the average of 30 experiments, and the results in bold are the best.

Table 2. Comparisons against our proposal

Function	Sombra et al. [12]	PSO from [11]	Proposal
F1	8.8518e-34	1.8e-3	**5.35e-40**
F2	1.1564e-10	2.0	**2.76e-19**
F3	468.4431	4.1e + 3	**7.61e-6**
F4	0.0912	8.1	**6.58e-11**
F5	61.2473	3.6e + 4	**9.56**
F6	0.1000	1.0e-3	**0**
F7	0.0262	0.04	**6.15e-4**
F8	-2.6792e + 3	-9.8e + 3	**-2.95e + 3**
F9	17.1796	55.1	**6.87**
F10	6.3357e-15	9.0e-3	**9.06e-25**
F11	4.9343	0.01	**4.82e-5**
F12	0.1103	0.29	**2.54e-4**
F13	0.0581	3.1e-18	**3.45e-43**
F14	2.8274	0.998	**1.27e-7**
F15	0.0042	2.8e-3	**4.66e-8**

5 Conclusions

The results of the comparisons show that the proposed approach can obtain on average better quality of the results, when compared with the original GSA method, and even with an improved GSA method. The fuzzy system used to control the parameters of GSA helps in the improvement of the performance of the algorithm, and the main parameters of GSA are α that affect the gravitational constant G, and *kbest* that allows

only some agents to apply their gravitational force to all of other agents. The use of a fuzzy system to model complex problems, like the parameter adaptation is an excellent choice, because it is easy to develop and a powerful tool when a way to model a problem is needed, and with the use of membership functions and fuzzy rules, most problems can be easily modeled. With the proposed methodology for parameter adaptation, now the GSA can be applied to different problems without the need of manually changing their parameters. The analysis of the parameters of GSA helps to know what parameters are the most important and have the greatest impact on the behavior of the algorithm. The proposed approach improves the quality of the results of GSA, and optimizes the behavior of the algorithm by deciding the best values for the parameters of GSA depending on the values of the inputs or the metrics of the status of GSA in each of the iterations.

References

1. Bahrololoum, A., Nezamabadi-pour, H., Bahrololoum, H., Saeed, M.: A prototype classifier based on gravitational search algorithm. Appl. Soft Comput. **12**(2), 819–825 (2012). Elsevier, Iran
2. Engelbrecht, A.P.: Fundamentals of Computational Swarm Intelligence. University of Pretoria, South Africa
3. Hassanzadeh, H.R., Rouhani, M.: A multi-objective gravitational search algorithm. In: Second International Conference on Computational Intelligence, Communication Systems and Networks (CICSyN), pp. 7–12. IEEE, Liverpool (2010)
4. Hatamlou, A., Abdullah, S., Othman, Z.: Gravitational search algorithm with heuristic search for clustering problems. In: 3rd Conference on Data Mining and Optimization (DMO), pp. 190–193. IEEE, Putrajaya (2011)
5. Holliday, D., Resnick, R., Walker, J.: Fundamental of Physic. Wiley, Hoboken (1993)
6. Kennedy, J., Eberhart, R.C.: Swarm Intelligence. Morgan Kaufmann, San Francisco (2001)
7. Mirjalili, S., Hashim, S.Z.M.: A new hybrid PSOGSA algorithm for function optimization. In: International Conference on Computer and Information Application (ICCIA), pp. 374 – 377. IEEE, Tianjin (2010)
8. Mirjalili, S., Hashim, S., Sardroudi, H.: Training feedforward neural networks using hybrid particle swarm optimization and gravitational search algorithm. Appl. Math. Comput. **218** (22), 11125–11137 (2012). Elsevier, Malaysia
9. Olivas, F., Valdez, F., Castillo, O.: A comparative study of membership functions for an interval type-2 fuzzy system used to dynamic parameter adaptation in particle swarm optimization. In: Castillo, O., Melin, P., Pedrycz, W., Kacprzyk, J. (eds.). SCI, vol. 547, pp. 67–78. Springer, Heidelberg (2014). doi:10.1007/978-3-319-05170-3_5
10. Pagnin, A., Schellini, S.A., Spadotto, A., Guido, R.C., Ponti, M., Chiachia, G., Falcao, A.X.: Feature selection through gravitational search algorithm. In: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 2052–2055 IEEE, Prague (2011)
11. Rashedi, E., Nezamabadi-pour, H., Saryazdi, S.: GSA: a gravitational search algorithm. Inf. Sci. **179**(13), 2232–2248 (2009). Elsevier, Iran
12. Sombra, A., Valdez, F., Melin, P., Castillo, O.: A new gravitational search algorithm using fuzzy logic to parameter adaptation. In: 2013 IEEE Congress on Evolutionary Computation (CEC), pp. 1068–1074. IEEE, June 2013

13. Verma, O.P., Sharma, R.: Newtonian gravitational edge detection using gravitational search algorithm. In: International Conference on Communication Systems and Network Technologies (CSNT), pp. 184–188. IEEE, Rajkot (2012)
14. Zadeh, L.: Fuzzy sets. Inf. Control **8**, 338–353 (1965)
15. Zadeh, L.: Fuzzy logic. IEEE Comput. Mag. **1**, 83–93 (1988)
16. Zadeh, L.: The concept of a linguistic variable and its application to approximate reasoning—I. Inform. Sci. **8**, 199–249 (1975)

Recurrent Flexible Neural Tree Model for Time Series Prediction

Marwa Ammar[1(✉)], Souhir Bouaziz[1], Adel M. Alimi[1], and Ajith Abraham[2,3]

[1] REsearch Groups in Intelligent Machines (REGIM), National School of Engineers (ENIS), University of Sfax, BP 1173, 3038 Sfax, Tunisia
marwa.ammar.tn@ieee.org
[2] Machine Intelligence Research Labs, Auburn, WA, USA
[3] IT4Innovations, VSB-Technical University of Ostrava, Ostrava, Czech Republic

Abstract. In this paper, a new encoding schemes based on tree representation is represented to encode recurrent multi layer neural network. It implement a learning process formed by two iterative phases: structure optimization and parameters optimization. For the structure evolving, a modified version of the Genetic Programming algorithm was adapted to support the recurrent topology of the network. On the other hand, a hybrid version of Harmony Search algorithm is used to adjust the network parameters including connection weights and neurons parameter set. Besides, the proposed model is evaluated by dynamical chaotic times series and compared with other studies.

Keywords: Recurrent Neural Networks · Tree encoding · Bi-objective optimization · Genetic Programming · Harmony Search · Dynamic time series

1 Introduction

Dynamic problems, which vary with time, have attracted the interest of many prediction research activities in a variety of areas such as pattern recognition, signal processing, robotics, and time series forecast. While time series problems have a direct impact on the human life (weather, solar, finance...), it is very useful to predict temporal sequences with accurate models. Recurrent Neural Network (RNN) can be regarded as suitable dynamic non linear system for confident prediction. RNN incorporate an internal memory retaining previous states of neurons to be useful in future iterations.

In general, the structure of RNN has an influence on its performance. Depending on the given problem, the RNN structure changes according to the number of neurons, number of layers and number of feed forward or feedback connections. Over the last decades, the tree representation was introduced to encode the feed forward Multi Layer Perceptrons [1]. Regarding to the performance arising of the flexible neural tree model, we adapt the tree encoding to the recurrent

A. Abraham et al. (eds.), *Proceedings of the 16th International Conference on Hybrid Intelligent Systems (HIS 2016)*, Advances in Intelligent Systems and Computing 552,
DOI 10.1007/978-3-319-52941-7_7

neural networks by defining nodes connection over different layers as parent-child relations.

In order to reach a good quality of RNN for prediction problems, it is primordial to establish a power learning technique. Currently, most of the satisfying learning techniques are based on intelligent heuristic algorithms called Evolutionary Computation algorithms. They are inspired from natural phenomena such as Genetic Algorithm (GA) [2], Particle Swarm Optimization (PSO) [3], Harmony Search (HS) [4], Artificial Bee Colony (ABC) [5], and so on.

In this work, the learning technique used for the recurrent flexible neural tree model includes two complementary optimization phase. First, an automated structure optimization was established to look for suitable network architecture for the given problem. It used a set of operations of the Genetic Programming algorithm for the recurrent network tree structure evolving. Second, a parameter optimization phase is needed for the weights connections and bias parameters adjustment.

In the next section, the adapted tree encoding for recurrent beta basis function neural network is detailed and the reasons that lead to relinquish other encoding schemes are presented. After that, the training process of the proposed model, including structure and parameter optimization, is described in Sect. 3. In Sect. 4, our model was applied to some benchmark chaotic times series to evaluates its performance. Finally, a concluding highlights are introduced in the conclusion section.

2 Tree Encoding of Recurrent Beta Basis Function Neural Network

Recurrent Neural Networks present the dynamic neural networks using inputs information as well as memorized neurons states. Therefore its performance depends widely on the network architecture. This complex architecture needs a suitable encoding to preserve useful information for optimization requirement. Some researchers assumed that separating the network (phenotype) from the genes (genotype) simplified the optimization task. In this context, two methods, direct encoding and indirect encoding, are used to convert the neural network into chromosome. However, both direct and indirect encoding schemes, are limited mainly by the combinatorial explosion of higher order neural networks dealing with big problem size. Hence, Zhang et al. [13] suggested an alternative encoding method based on the tree representation.

Tree encoding allows a direct manipulation of the structure without coding-decoding intervention. Moreover, neural tree could undergo efficiently a learning process of both structures and parameters using genetic search. Its configuration depends on the maximum depth, the maximum branches for each node (tree degree), the set of leaf nodes (terminal nodes) and the set of non leaf nodes (functional nodes). Neural tree connections are defined as a direct link (i, j) from parent $node_i$ to child $node_j$ and are associated with synaptic weights. Tree encoding applied successfully to the feed forward neural network [6,7].

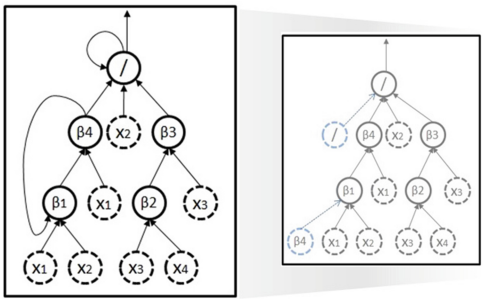


Fig. 1. The typical representation of the recurrent flexible beta basis tree with 4 inputs, 2 hidden layer, and 4 functional nodes

In this work, we proposed the tree representation to encode the recurrent network. The recurrent flexible neural tree model involves four layers; input layer including leaf nodes, two hidden layers enclosing leaf and non leaf nodes, and output layer for the output non leaf node. Leaf nodes correspond to the data inputs. However non leaf nodes are the functional nodes presenting the neurons of our network. While neural tree evaluation occurs from inputs nodes to output node, neural tree generation gets start from the output node, associating a random number of children for each parent, until reaching the terminals layer. The parent-child relation is adapted to represent feed forward and recurrent neural network connections. Indeed, we extend the set of feasible children of each parent nodes (non leaf node) to encompass not only different nodes from lower layers (feed forward connections) but also nodes in upper layers (feedback connections) and even the node itself (loop connection). Therefore, an internal memory associated in every functional node was integrated to memorize the node state used for the next iteration. A typical representation of the model is illustrated in Fig. 1.

Functional nodes were activated by the Beta function introduced in [14]. Beta equation, presented in Eq. 1, was adjusted by the strict positive values of p and q, the center c and the width d.

$$\beta(x,c,\sigma,p,q) = \begin{cases} \left[1+\frac{(p+q)(x-c)}{\sigma p}\right]^p \left[1-\frac{(p+q)(x-c)}{\sigma p}\right]^q, \\ if x \in \left]c-\frac{\sigma p}{p+q}, c+\frac{\sigma p}{p+q}\right[\\ 0, else \end{cases} \tag{1}$$

3 Evolving the Recurrent Flexible Beta Basis Function Neural Tree Model

After generating a random population of Recurrent Flexible Beta basis function Neural Trees, an automated evolving process of both structure and parameters was trigged to provide the optimal or the near optimal solution for the given problem.

3.1 Structure Evolving Based Genetic Programming

Over the past decades, a strong relation has been implemented between neural networks and genetic algorithms. It could be due to the natural genetic construction of biological neural networks. In fact, the use of genetic algorithms in adapting neural networks has three shapes: (1) set connection weights in fixed structures, (2) automatic search of best neural network structures and (3) select training data for training networks. In this work, the second deal was established to learn the recurrent flexible beta basis function neural tree topologies.

Using the tree encoding, training and fitness evaluation is not preceded by genotype-phenotype decoding. Genetic Programming (GP) algorithm [15] adapted genetic operators, such as crossover and mutation, to be applied directly on the neural trees structures. GP was efficiently applied to a broad class of feed forward neural trees [1,16]. A modified variant of Genetic Programming algorithm was proposed to evolve the recurrent neural trees topologies. This algorithm incorporate three main operators:

Evaluation and Selection: It is a preliminary phase for both crossover and mutation operators in which two parents have been selected from the current population. The selection was based on two fitness functions measuring the structure complexity and the accuracy of the model. Structure complexity function f_1 was determined by the number of functional nodes in the network and its distribution in the hidden layers (see Eq. 2). For the model accuracy, the Normalised Mean Square Error (see Eq. 3) was chosen to compute the prediction error of the model (f_2). In order to solve the trade off between the two objective functions, a multi objective selection strategy was adapted. It classify individuals in current population in a number of fronts according to the dominance relation with applying the non domination sort algorithm [17]. Then, for N resulted fronts, two parents will be selected from the $(N-1)$ survived fronts. Solutions in the last front will be removed and replaced by new recurrent neural trees. Moreover, a binary tournament selection was applied to select between crossover and mutation operators for the designed parents. Each offspring will be selected in the next generation if it dominate at least one of its parents.

$$(f_1) : SC = \frac{\sum_{i=1}^{LN} path(node_i)}{LN} * NLN \tag{2}$$

where LN and NLN are the number of leaf nodes and non leaf nodes respectively.

$$(f_2) : NMSE = \frac{\sum_{j=1}^{p} (y_j^t - y_j^{out})^2}{\sum_{j=1}^{p} (y_j^t - \bar{y}_j)} \tag{3}$$

where p represents the samples number, y_t, y_{out} are the desired and the predicted output, and $\bar{y}$ is the average of desired outputs.

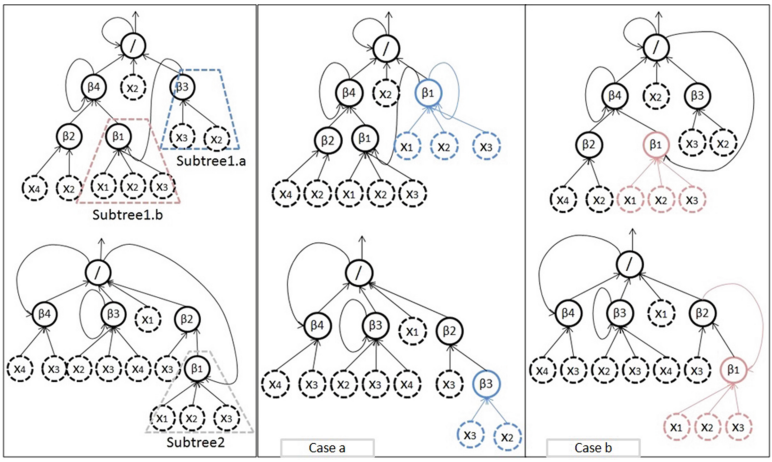


Fig. 2. Examples of crossover operator on RFBNTs

Crossover: It is a partial combination between selected parents or recurrent neural trees. It permutes two sub trees, including weights and Beta parameters, to provide two new trees. The sub tree was determined by cutting a connection between two functional nodes in the network. The crossover between two recurrent neural trees is more difficult on account of different existing connections, feed forward and feedback. In feed forward connections, a hierarchic connections related parents with their children.

However, recurrent connections could link a parent with itself, its parent, the parent of its parent, or one child of the parent of its parent. These relations are strictly related to the specific network topology. For that, in the crossover interchange, only feasible connections will be totally conserved. For non feasible recurrent connection, the recurrent child will be replaced by a possible one, from the new recurrent neural tree, without changing the connection weight.

In fact the disharmony in recurrent connections could be exist in some cases:

- parent of its parent: if the node ascend in the upper hidden layer in the new tree.
- child(i) of the parent of its parent: if (i > number of child of the parent of its parent) or (the designed child(i) is a leaf node)

The Fig. 2 illustrate two examples of crossover operator for two recurrent neural trees. We chose to present the two crossover disharmony case through two different cases (a and b). In case a, the 'parent-of-parent' recurrent connection of $\beta 1$ in subtree 2 could not take place after the crossover. So it was replaced by a random one (blue connection). In case b, the 'child(3)-of-parent' recurrent connection of $\beta 1$ in subtree 1.b meet a problem after the crossover step when it will correspond to a leaf node in the new tree. Then, it was replaced by a random recurrent connection with red color.

Mutation: It is a structural transformation of the parent to result a new offspring. It could be performed with three operators:

- changing one leaf node: a leaf node, selected randomly, is replaced by another one.
- changing all leaf nodes: all leaf nodes in the tree should be changed
- growing: when a random selected sub tree is replaced by a new generated sub tree having the same or a smaller depth value. In this case, some disharmonies in recurrent connection could be faced.

The global Modified Genetic Programming process is an iterative algorithm limited by two stopping criteria. The first controls the number of iteration for not exceed the maximum allowed number. The second verifies if the best solution, which made the minimal error, has reached the fixed optimum.

3.2 Parameter Evolving Based Harmony Search

After the structure evolving phase, the RFBNT model require an adjustment of its parameters in order to improve its accuracy. These parameters include connection weights and Beta parameters such as the centre c, the width d, p and q. The optimization of the set of parameters is a hard task which need a powerful algorithm. Since its appearance, Evolutionary Computation algorithms have been attracted the researchers attention and admiration. Among of them, the Harmony Search algorithm introduced by Zong Woo Geem in 2001 [4]. It was inspired from the improvisation in jazz music. When a musician is looking for a nice harmony between notes, it should optimize several sound waves with different frequencies. An hybridized variant of Harmony Search algorithm with Particle Swarm Optimization (PSO) algorithm [3], called IHSPSO, is used to adjust the model parameters. More details about HSPSO algorithm are in stored in [19].

4 Experimental Results

A set of Benchmark dynamical non linear time series including Henon map and Rossler attractor, are disposed to evaluate our model performance. For that, we need first to set up the RFBNT model by initializing some parameters (see Table 1).

Henon map: It is a chaotic discrete time series and it is usually used to test dynamical system. Henon map time series [20] is generated through two equations written as:

$$\begin{aligned} x_{n+1} &= y_n + 1 - \alpha x_n^2 \\ y_{n+1} &= \beta x_n \end{aligned} \tag{4}$$

where $\alpha = 1.4$ and $\beta = 0.3$.

Table 1. Parameter setting

Modified GP		IHSPSO	
Population size	50	PARmin, PARmax	0.00001, 1.0
Crossover probability	0.3	BWmin, BWmax	0.00001, 1.0
Mutation probability	0.6	HMCR, NI	0.9, 50
Generation gap	0.9	(c1, c2)	(0.2, 0.7)

Rossler attractor: It was introduced by Otto Rossler [21] as a continuous time system for generating chaotic dynamic time series. It was described by three ordinary differential equations:

$$\begin{aligned} \dot{x} &= -y - z \\ \dot{y} &= x + ay \\ \dot{z} &= b + z(x - c) \end{aligned} \tag{5}$$

where a = 0.2, b = 0.2, and c = 4.6.

For both data sets, we implement a data set of 1600 time steps divided into training set (800 samples) and testing set (800 samples). A set of delayed input features x(n − 1), x(n − 2), x(n − 3) and x(n − 4) are used in order to predict the coming output y(n). Figures 3 and 4 plot the predicted output of the RFBNT model for the Henon and Rossler datasets respectively. These results are promising and report the ability of the model in learning dynamic behaviours.

Different measures, including NMSE and learning time, are used to evaluate the model in term of accuracy and time complexity. These results are

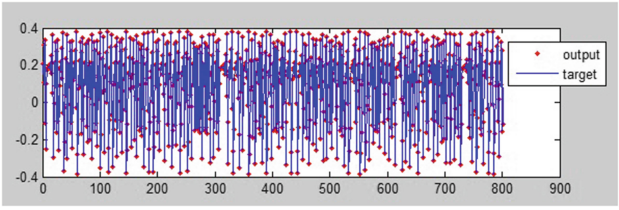


Fig. 3. Target and predicted output for Henon map time series

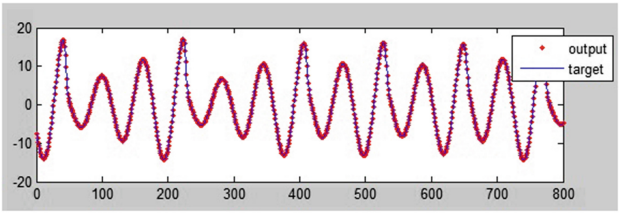


Fig. 4. Target and predicted output for Rossler attractor time series

Table 2. Prediction results of RFBNT model for Henon map and Rosslor attractor times series over 10 runs compared with other methods

Method	Henon map			Rossler attractor		
	NMSE tr	NMSE ts	Time (s)	NMSE tr	NMSE ts	Time (s)
MLP-EKF [22]	2.3 e-04	1.62 e-03	17.77	2.5 e-04	1.93 e-03	15.45
MLP-BLM [22]	3.3 e-04	9.60 e-04	14.71	4.7 e-04	1.01 e-03	12.78
RNN-BPTT [22]	5.6 e-04	1.85 e-03	21.28	7.0 e-04	3.11 e-03	26.88
RNN-RTRL [22]	5.7 e-04	1.72 e-03	29.62	7.1 e-04	3.12 e-03	25.76
RNN-EKF [22]	3.6 e-04	1.21 e-03	33.04	6.0 e-04	1.91 e-03	28.72
RBLM-RNN [22]	3.6 e-04	9.00 e-04	19.83	5.7 e-04	9.2 e-04	15.02
LLNF [23]	1.3 e-04	2.90 e-04	–	4.8 e-05	7.1 e-05	–
LNF [23]	1.7 e-05	6.40 e-05	0.31	6.5 e-06	1.5 e-03	0.31
LoLiMoT_RBF [24]	–	3.60 e-10	–	–	–	–
DLE-VQIT, [25]	–	1.92 e-02	–	–	–	–
TDL-MLP [25]	–	7.59 e-02	–	–	–	–
RFBNT	**6.42 e-11**	**6.50 e-11**	**Str: 9.97 Par:29.94**	**1.81 e-06**	**6.87 e-06**	**Str: 7.73 Par:10.85**

compared with other models in the literature (see Table 2) which belong to different neural network topologies and different learning algorithms. Our model exhibit best results in prediction accuracy. Indeed, presented results are far better than results provided from different models such as feed forward neural networks [22,25], recurrent neural networks [22], and Neuro-Fuzzy models [23] in most prediction tests. Regarding to the spent time in learning our model, it seem to be slightly higher than other models. It is explained by the fact that our model operates two learning phases, iteratively, for structure and parameters optimization. However, the best model solution for predicting Henon and Rossler chaotic time series implement only 4 and 3 neurons, respectively, distributed in the two hidden layers (see Fig. 5). So, it's worth it. This study demonstrates the superior performance of RFBNT due to the flexible tree coding and the powerful training algorithms of structure and parameters of the model.

Table 3. Influence of noise on RFBNT performance for Henon and rossler time series

Noisy data	0.01	0.02	0.03	0.04	0.05
Henon map	2.25 e-05	9.52 e-05	2.26 e-04	4.23 e-04	6.92 e-04
Rossler attractor	9.21 e-04	3.76 e-04	1.30 e-03	3.82 e-03	5.11 e-03

Then, training data sets are distorted by a white noise power varying from 0.01 to 0.05. The NMSE was computed to the model predicting noisy systems (see Table 3). It was extended from 2.25 e-05 to 6.92 e-04 with Henon map and from 9.21 e-04 to 5.11 e-03 with Rossler attractor.

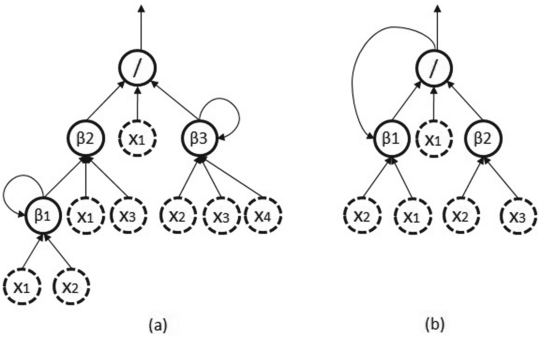


Fig. 5. The RFBNT model for Henon map (a) and Rossler attractor(b) time series

5 Conclusion

In this work, a Recurrent Flexible Beta basis function Neural Tree (RFBNT) model is presented. It is a recurrent multi layer neural network using the flexible tree representation to encode its architecture. In addition, two powerful algorithms are used for training the RFBNT: Modified Genetic Programming is introduced for structure evolving and Hybrid Harmony Search algorithm is used for parameters adjustment. We evaluate the model performance by applying Henon map time series and Rossler attractor time series and compared its obtained prediction results with other models.

References

1. Chen, Y., Abraham, A.: Tree-Structure Based Hybrid Computational Intelligence: Theoretical Foundations and Applications, vol. 2. Springer Science and Business Media, New York (2009)
2. Holland, J.H.: Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence (1975)
3. Kennedy, J., Eberhart, R.: Particle swarm optimization. In: IEEE International of First Conference on Neural Networks
4. Geem, Z.W., Kim, J.H., Loganathan, G.: A new heuristic optimization algorithm: harmony search. Simulation **76**(2), 60–68 (2001)
5. Karaboga, D.: An idea based on honey bee swarm for numerical optimization Technical report-tr06, Erciyes University, engineering faculty, computer engineering department, Technical report (2005)
6. Bouaziz, S., Dhahri, H., Alimi, A.M., Abraham, A.: A hybrid learning algorithm for evolving flexible beta basis function neural tree model. Neurocomputing **117**, 107–117 (2013)
7. Chen, Y., Yang, B., Dong, J., Abraham, A.: Time-series forecasting using flexible neural tree model. Inf. Sci. **174**(3), 219–235 (2005)
8. Hussain, T.S., et al.: Genetic encoding of neural networks using attribute grammars

9. Fekiavc, J., Zelinka, I., Burguillo, J.C.: A review of methods for encoding neural network topologies in evolutionary computation. In: Proceedings of 25th European Conference on Modeling and Simulation ECMS 2011, pp. 410–416 (2011)
10. Prusinkiewicz, P., Lindenmayer, A.: The algorithmic beauty of plants (the virtual laboratory)
11. Gruau, F., Whitley, D.: Adding learning to the cellular development of neural networks: evolution and the Baldwin effect. Evol. Comput. **1**(3), 213–233 (1993)
12. Kitano, H.: ΦDmDialog: a speech-to-speech dialogue translation system. Mach. Transl. **5**(4), 301–338 (1990)
13. Zhang, B.-T., Ohm, P., Muhlenbein, H.: Evolutionary induction of sparse neural trees. Evol. Comput. **5**(2), 213–236 (1997)
14. Alimi, A.M.: The beta fuzzy system: approximation of standard membership functions. In: Proceedings of the 17eme Journees Tunisiennes d'Electrotechnique et d'Automatique: JTEA 1997, pp. 108–112 (1997)
15. Koza, J.R.: Genetic programming: a paradigm for genetically breeding populations of computer programs to solve problems. Stanford University, Department of Computer Science (1990)
16. Bouaziz, S., Dhahri, H., Alimi, A.M.: Evolving flexible beta operator neural trees (FBONT) for time series forecasting. In: Huang, T., Zeng, Z., Li, C., Leung, C.S. (eds.) ICONIP 2012. LNCS, vol. 7665, pp. 17–24. Springer, Heidelberg (2012). doi:10.1007/978-3-642-34487-9_3
17. Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Trans. Evol. Comput. **6**(2), 182–197 (2002)
18. Mahdavi, M., Fesanghary, M., Damangir, E.: An improved harmony search algorithm for solving optimization problems. Appl. Math. Comput. **188**(2), 1567–1579 (2007)
19. Ammar, M., Bouaziz, S., Alimi, A.M., Abraham, A.: Hybrid harmony search algorithm for global optimization. In: 2013 IEEE World Congress on Nature and Biologically Inspired Computing (NaBIC), pp. 69–75 (2013)
20. Henon, M.: A two-dimensional mapping with a strange attractor. Commun. Math. Phys. **50**(1), 69–77 (1976)
21. Rossler, O.: An equation for hyperchaos. Phys. Lett. A **71**(2), 155–157 (1979)
22. Mirikitani, D.T., Nikolaev, N.: Recursive Bayesian recurrent neural networks for time-series modeling. IEEE Trans. Neural Networks **21**(2), 262–274 (2010)
23. Miranian, A., Abdollahzade, M.: Developing a local least-squares support vector machines-based neuro-fuzzy model for nonlinear and chaotic time series prediction. IEEE Trans. Neural Netw. Learn. Syst. **24**(2), 207–218 (2013)
24. Gholipour, A., Araabi, B.N., Lucas, C.: Predicting chaotic time series using neural and neurofuzzy models: a comparative study. Neural Process. Lett. **24**(3), 217–239 (2006)
25. Martinez-Rego, D., Fontenla-Romero, O., Alonso-Betanzos, A.: A method for time series prediction using a combination of linear models. In: Advances in Computational Intelligence and Learning

Multiple Criteria Inventory Classification Approach Based on Differential Evolution and Electre III

Hedi Cherif[1(✉)] and Talel Ladhari[2,3]

[1] Institut Supérieur de Gestion, Université de Tunis, Tunis, Tunisia
cherif.hedi@gmail.com
[2] Ecole Supérieure des Sciences Economiques et Commerciales de Tunis, Université de Tunis, Tunis, Tunisia
[3] College of Business, Umm Al-Qura University, Umm Al-Qura, Mecca, Saudi Arabia

Abstract. The ABC classification represents one of the most frequently used analysis in production and inventory management domains. This analysis is applied to categorize a set of items in three predefined classes A, B and C, where each class follows a specific management and control policies, in order to generate companies financial well-being. This paper introduces a new approach for the multi-criteria inventory classification based on the hybridization of the Differential Evolution algorithm (DE) with the multi-criteria decision making method namely Electre III. The evolutionary algorithm (DE) attends to learn and optimize the Electre III input parameters (criteria weights). The Electre III method generates a ranking score for all the inventory items and an ABC distribution dispatches all these items into three ordered classes A, B, C, forming a complete classification. An inventory cost function is used thereafter to evaluate each established classification. This function is based on different inventory costs and service level measurement and also represents the objective function of our model, which consists of minimizing the inventory cost. The highlight of our proposed hybridization approach DE-Electre III is the exploitation of the robustness and efficiency of used techniques. Based on generated results, our model provided encouraging results in the ABC MCIC problem.

Keywords: Inventory management · ABC multi-criteria inventory classification · Hybrid model · Differential Evolution · Electre III

1 Introduction

The classification of inventory is a way or more explicitly a strategy in how that inventory will be managed. The most popular and a widely used approach for the inventory classification problem is the ABC analysis, which consists of classifying each inventory item to one group among A, B, or C groups. Each group has an appropriate levels of control. To deal with MCIC problems, various

A. Abraham et al. (eds.), *Proceedings of the 16th International Conference on Hybrid Intelligent Systems (HIS 2016)*, Advances in Intelligent Systems and Computing 552,
DOI 10.1007/978-3-319-52941-7_8

methodologies have been used in literature to categorize inventory items by taking into account their computed scores, such as Artificial Intelligence, Multi-Criteria Decision Making and the Mathematical Programming. A ranking of these inventory items is established based on a score for each item, which is calculated by combining the evaluations of the items on the different criteria with the weights of the criteria, according to an aggregation function. Using this items ranking, ABC analysis places the items into three ordered classes A, B and C, according to a predefined class distribution.

ABC analysis has been widely studied in literature and especially in the area of inventory management. The Analytic Hierarchy Process (AHP) method, developed by Saaty [24], is among the most used methods in multi-criteria inventory classification. Since, several studies e.g. [3,7,8,19,20] have used the AHP method to solve the MCIC problem in different manners. Other studies are based on fuzzy AHP (FAHP) in order to incorporate the decisions of the decision makers and find the criteria weights [12–14].

Metaheuristics and genetic algorithms were involved in resolving the MCIC problem. Guvenir and Erel [9] developed first a method that use the generic algorithm in order to learn criteria weight, and have established cut-off points between the classes A–B and B–C, in order to generate a classification of items. Then they showed in a second study [10] that their method based on genetic algorithm give better performance than the AHP method. Tsai and Yeh [27] use the particle swarm optimization technique and present an inventory classification algorithm that simultaneously searches the optimum number of inventory classes and perform classification, while Mohammaditabar et al. deploys the simulating annealing method [17] and proposes an integrated model to categorize the items and at the same time find the best policy.

As for Multi-Criteria Decision Making methods, they have been used to improve the inventory classification. Bhattacharya et al. [2] propose a model which combines Topsis (Technique for Order Preferences by Similarity to the Ideal Solution) with AHP method. Chen et al. [5] propose an alternative approach to MCIC problem by using Topsis and two virtual items. Liu et al. [15] combines the methods of Electre III and the simulating annealing to deal with the compensatory effect of the items against criteria and opted for grouping criteria.

Linear and nonlinear optimization models have been proposed in literature in order to generate a weight vector that optimizes the weighted score of each item. Ramanathan [21] has developed a weighted linear optimization model (called R model) that uses a weighted additive function in order to generate a set of optimal weights. These criteria weights maximize the score of each item. This R model was further improved by Zhou and Fan [28] proposed a ZF-model to address deficiencies and subjectivity of R model, by using two sets of weights that are most favourable and least favourable for each item. Ng [18] developped another linear optimization model called Ng model and Hadi [12] implemented the H model as a nonlinear optimization model, which computes the optimal items score by keeping the weight effects. Chen [4] provides a peer-estimation

approach based on two weighted linear optimization models R and ZF, which aggregate two performances scores generated from the models R and ZF.

To the best of our knowledge, very few researches have been proposed so far to combine MCIC methods in order to improve the task of inventory managers. This literature deficiency compared to multi-criteria inventory classification hybridization methods has led us to focus in this paper on combination of different techniques from different families of methods and develop new hybrid models to better evaluate the inventory classification.

The main contribution of this paper is the development of a new hybrid approach based on DE method [25,26] with a Multi-Criteria Decision Making method, namely Electre III [6,22,23], to tackle the ABC inventory classification problem. In other words, we use the DE method to generate solutions within the constraints of the problem. These solutions represent the weights of the criteria and serve as input parameters by Electre III method, which have the task to generate an items ranking, based on score. An ABC distribution will be subsequently assigned to these items to dispatch them between the A, B and C classes, in order to obtain an ABC classification of the solution.

The rest of this paper is organized as follows. In Sect. 2, we briefly present the used methods of our hybrid models (DE and Electre III). The proposed hybrid optimization model is described in Sect. 3. Section 4 presents the dataset, the experimental results, and comparative numerical studies with some models from the literature. The paper ends with conclusions and discussion regarding future research.

2 Definitions and Basic Concepts

2.1 Differential Evolution

The differential evolution method [25,26] is probably one of the most powerful stochastic optimization algorithms. This evolutionary algorithm uses NP vectors of D dimensions as the population at each generation G. The steps of DE method are as follows:

- **Initialization:** The vectors of the initial population are randomly generated and cover the entire search space, where each parameter of the initial vector is generated from the following equation:

$$x_{j,i,0} = x_{j,min} + rand_{i,j}[0,1](x_{j,max} - x_{j,min}) \tag{1}$$

  Where $x_{j,min}$ and $x_{j,max}$ are respectively the lower and upper bounds of the search space and $rand_{i,j}[0,1]$ is a uniformly distributed random number $\in [0,1]$.
- **Mutation:** a mutant vector $v_{i,G}$ is generated for each target vector $x_{i,G}$, according to:

$$v_{i,G+1} = x_{r1,G} + F(x_{r2,G} - x_{r3,G}) \tag{2}$$

$r1$, $r2$ and $r3$ are randomly chosen indexes, belonging to the interval $\{1, 2, \ldots, NP\}$, mutually different and must be different from the current index i. F represent a real in $\in [0, 2]$ which has an effect on the differential variation amplification $(x_{r2,G} - x_{r3,G})$. By convention, the factor F will be set to the value 0.5.

- **Crossover:** this step is useful for the diversification of vector parameters. For this, each element of the trial vector $u_{i,G+1}$ is generated as follows:

$$u_{ji,G+1} = \begin{cases} v_{ji,G+1} \text{ if } (rand(j) \leq P_{CR}) \text{ or } j = rnbr(i) \\ x_{ji,G} \quad \text{if } (rand(j) > P_{CR}) \text{ and } j \neq rnbr(i) \end{cases} \tag{3}$$

With $j \in [1, D]$, $rand(j)$ is a random number generated in $[0, 1]$, P_{CR} is the crossover probability $\in [0, 1]$. For ensure that the trial vector $u_{i,G+1}$ receives at least one parameter from the mutant vector $v_{i,G+1}$, we choose a random index $rnbr(i)$ between 1 and D.

- **Selection:** This step indicates if the target vector will be maintained in the next generation or replaced by the trial vector, with $f(x)$ representing the objective function to be minimized.

$$x_{i,G+1} = \begin{cases} u_{i,G} \text{ if } f(u_{i,G}) \leq f(x_{i,G}) \\ x_{i,G} \text{ if } f(u_{i,G}) > f(x_{i,G} \end{cases} \tag{4}$$

2.2 Electre III

Electre III is a MCDM method [6,22,23] that uses the pairwise comparison in order to categorize a collection of items, taking into account an immeasurable and conflicting criteria. The method consists of two steps: In the first step, a valued outranking relation is constructed for each pair of items A and B, by determining the degree of credibility of the statement that "A is at least as good as B" or although "A outperforms B". In the second step, these valued outranking relations are used to generate a ranking of all items.

To measure the degree of credibility of the statement "a outranks b", denoted by $\sigma(a, b)$, four steps must be followed:

- **Step 1:** computation of partial concordance indices for each pair of items a and b:

$$C_j(a, b) = \begin{cases} 0 & \text{if } g_j(b) - g_j(a) \geq pj \\ 1 & \text{if } g_j(b) - g_j(a) \leq qj \\ \frac{pj + g_j(a) - g_j(b)}{pj - qj} & \text{otherwise.} \end{cases} \tag{5}$$

Where $g_j(a)$ represent the evaluation of the item a according to the criterion j, q_j is the indifference threshold and pj represents the preference threshold. The indifference threshold is equal to the biggest difference between two evaluations according to the same criterion, for which the decision-maker is unable to make a clear choice. Preference threshold is the smallest difference between two evaluations on the same criterion, for which the decision-maker is able to make a clear preference for an item.

- **Step 2:** computation of the global concordance index $C(a,b)$ for each pair of items a and b:

$$C(a,b) = \frac{\sum_{j=1}^{n} w_j \times C_j(a,b)}{\sum_{j=1}^{n} w_j} \tag{6}$$

With w_j represent le weight of criterion j. The global concordance index $C(a,b)$ is a measure of the strength of arguments that associate with the statement "a outranks b".

- **Step 3:** computation of partial discordance indices $D(a,b)$ for each pair of items a and b:

$$D_j(a,b) = \begin{cases} 0 & \text{if } g_j(b) - g_j(a) \leq pj \\ 1 & \text{if } g_j(b) - g_j(a) \geq vj \\ \frac{g_j(b) - g_j(a) - pj}{vj - pj} & \text{otherwise.} \end{cases} \tag{7}$$

Where v_j is the veto threshold which represents the tolerance limit that can accept makers for compensation. The partial discordance indice $D(a,b)$ represent the measure of the strength of arguments that disagree with the statement "a outranks b" according to the criterion j.

- **Step 4:** computation of the credibility index of the statement "a outranks b":

$$\sigma(a,b) = \begin{cases} C(a,b) \text{ if } D_j(a,b) \leq C(a,b)\ \forall j \\ C(a,b) \times \prod\limits_{D_j(a,b) > C(a,b)} \frac{1 - D_j(a,b)}{1 - C(a,b)} \end{cases} \tag{8}$$

The credibility index corresponds to the concordance index weakened by the possible effects of veto.

Since the exploitation of the outranking relation is difficult to understand by decision-makers because of its complexity (Preorder construction, distillation phase and intersection of preorders), we will use the exploitation of the outranking relation of PROMETHEE II method [16] in order to generate the global score of each element, using the degree of credibility $\sigma(a,b)$ calculated in the previous step. Therefore, to calculate the overall score of each item we proceed in these steps:

- **Step 5:** computation of the positive and negative outranking flow:

$$\Phi^{+}(a) = \frac{1}{m-1} \sum_{x \neq a} \sigma(a,x) \tag{9}$$

$$\Phi^{-}(a) = \frac{1}{m-1} \sum_{x \neq a} \sigma(x,a) \tag{10}$$

m is the number of total items. The positive outranking flow $\Phi^{+}(a)$ expresses how an item outperforms all other elements, in other words, it represents the

power of an element, while the negative outranking flow $\Phi^-(a)$ represents how an item is preceded by all other elements. It is considered the weakness of an item.

- **Step 6:** computation of the net outranking flow:

$$\Phi(a) = \Phi^+(a) - \Phi^-(a) \tag{11}$$

The net outranking flow is used to generate a complete ranking of the items.

3 A New Hybrid Approach for ABC MCIC

Our new hybrid model DE-Electre III uses in the first step the DE method as a generator of solutions that respect the constraints imposed by the MCDM methods. Each generated solution by DE method represent in fact a criteria weights vector. The learning process of our model is provided by DE method. The MCDM method deployed in our hybrid model (Electre III) uses each time a solution generated by DE method (weight vector) as an input parameter. The Electre III method aims to generate a score for each item. After calculating all the scores, an items ranking is made, sorting by the scores in decreasing order. Then, the ABC classification approach categorizes inventory items into three groups labeled A (very important), B (moderately important) and C (least important). To compare objectively all the performance of the optimization models, an estimation function based on the inventory cost and the fill rate service level is used [1], in order to evaluate the item classifications of each model. By setting the specified service level for each class (0.99 for class A, 0.95 for class B and 0.9 for class C), this inventory performance evaluation method estimates two important measures: the total holding inventory cost for all items and the achieved overall service level of the system. To present this performance evaluation method, we use the same notation as in [1].

The total safety stock inventory cost is given by:

$$C_T = \sum_{i=1}^{N} h_i k_i \sigma_i \sqrt{L_i} \tag{12}$$

For each inventory item i, the safety factor k_i is calculated as follow:

$$k_i = \Phi^{-1}(CSL_i) \tag{13}$$

The Fill Rate of each item i can be approximated by:

$$FR_i = 1 - \frac{\sigma_i \sqrt{L_i}}{Q_i} G(k_i) \tag{14}$$

Where

$$G(k_i) = \frac{1}{\sqrt{2\pi}} e^{-\frac{k_i^2}{2}} (1 - \Phi(k_i)) \tag{15}$$

The overall FR of the inventory system is calculated as follows:

$$FR_T = \frac{\sum_{i=1}^{N} FR_i D_i}{\sum_{i=1}^{N} D_i} \tag{16}$$

4 Computational Results

4.1 Adaptation Methods

The DE method must generate solutions that respect two conditions: (a) the values of the criteria weights generated must be between 0 and 1 and (b) the sum of all weights must be equal to 1. Accordingly, we adapt the DE method to achieve these two conditions, as follows:

$$x_{i,j} = x_{max} - rand\left[0, \left[x_{max} - \sum_{t=1}^{D} x_{i,t}\right]\right] \tag{17}$$

Where $x_{max} = 1$ (upper bound of the research space) and D the total of criteria.

The mutation step of the DE method (Eq. 2) generates a mutant vector which can have values outside the lower and upper bounds of the search space (respectively 0 and 1). To address this conflict, we calibrated the values so that the parameters remain within the search space:

$$x_{i,j} = \begin{cases} 0 & \text{if } x_{i,j} < 0 \\ 1 & \text{if } x_{i,j} > 1 \\ x_{i,j} & \text{otherwise.} \end{cases} \tag{18}$$

However, this calibration does not ensure that the sum of the values vector is equal to 1, which represents the sum of criteria weights. For this purpose, we made the vector normalization in order to this sum must be equal to 1, as following:

$$x_{i,j} = \frac{x_{i,j}}{\sum_{t=1}^{D} x_{i,t}} \tag{19}$$

4.2 Experimental Results

To evaluate the performance of our proposed hybrid model in the ABC inventory classification context, we consider the data set provided by a Hospital Respiratory Therapy Unit (HRTU). This data set which contains 47 inventory items evaluated on three criteria (Annual Dollar Usage (ADU), Average Unit Cost (AUC) and Lead Time (LT)) is used by many researchers [4,11,18,21,28].

Table 1 lists the evaluations of the 47 inventory items with the three criteria, as well as the results obtained by our hybridization model DE-Electre III with the existing models from the literature. All the classifications have the same ABC distribution which dispatches 10 items in class A, 14 items in class B and 23 items in class C. The best classification cost belongs to our proposed model DE-Electre III model with a cost equal to 822.417, against 927.517 for the R model which represents the best cost among all the existing models. All these classifications have been established with a good Fill Rate (all the rates are higher than 0.97), reflecting a good classification and a customer satisfaction.

Table 1. Classification of DE-Electre III vs existing models.

Item	ADU	AUC	LT	R [21]	ZF [28]	Chen [4]	H [11]	NG [18]	Our model
1	5840.64	49.92	2	A	A	A	A	A	C
2	5670	210	5	A	A	A	A	A	A
3	5037.12	23.76	4	A	A	A	A	A	C
4	4769.56	27.73	1	B	C	B	A	A	C
5	3478.8	57.98	3	B	B	B	A	A	C
6	2936.67	31.24	3	C	C	B	B	A	C
7	2820	28.2	3	C	C	B	B	B	C
8	2640	55	4	B	B	B	B	B	B
9	2423.52	73.44	6	A	A	A	A	A	A
10	2407.5	160.5	4	B	A	A	A	A	B
11	1075.2	5.12	2	C	C	C	C	C	C
12	1043.5	20.87	5	B	B	B	B	B	B
13	1038	86.5	7	A	A	A	A	A	A
14	883.2	110.4	5	B	A	B	A	B	A
15	854.4	71.2	3	C	C	C	C	C	B
16	810	45	3	C	C	C	C	C	C
17	703.68	14.66	4	C	C	C	C	C	C
18	594	49.5	6	A	A	B	B	B	A
19	570	47.5	5	B	B	B	B	B	B
20	467.6	58.45	4	C	B	C	C	C	B
21	463.6	24.4	4	C	C	C	C	C	C
22	455	65	4	C	B	C	C	C	B
23	432.5	86.5	4	C	B	C	B	B	B
24	398.4	33.2	3	C	C	C	C	C	C
25	370.5	37.05	1	C	C	C	C	C	C
26	338.4	33.84	3	C	C	C	C	C	C
27	336.12	84.03	1	C	C	C	C	C	C
28	313.6	78.4	6	A	A	A	B	B	A
29	268.68	134.34	7	A	A	A	A	A	A
30	224	56	1	C	C	C	C	C	C
31	216	72	5	B	B	B	B	B	A
32	212.08	53.02	2	C	C	C	C	C	C
33	197.92	49.48	5	B	B	B	B	B	B
34	190.89	7.07	7	A	B	A	B	B	B
35	181.8	60.6	3	C	C	C	C	C	C
36	163.28	40.82	3	C	C	C	C	C	C
37	150	30	5	B	B	B	C	C	B
38	134.8	67.4	3	C	C	C	C	C	C
39	119.2	59.6	5	B	B	B	B	B	B
40	103.36	51.68	6	B	B	B	B	B	A
41	79.2	19.8	2	C	C	C	C	C	C
42	75.4	37.7	2	C	C	C	C	C	C
43	59.78	29.89	5	B	C	C	C	C	B
44	48.3	48.3	3	C	C	C	C	C	C
45	34.4	34.4	7	A	B	A	B	B	A
46	28.8	28.8	3	C	C	C	C	C	C
47	25.38	8.46	5	B	C	C	C	C	B
Classification cost				927.517	945.357	958.143	999.892	1011.007	**822.417**
Fill rate				0.986	0.984	0.988	0.99	0.991	0.972

5 Conclusion

We propose in this paper a new hybrid model to address the ABC MCIC problem. To obtain the optimal classification, we combine the Differential Evolution method with the method Electre III to solve such a combinatorial optimization problem. Our effective hybrid model starts by learning criteria weights using DE method and theses weights are subsequently recovered by the MCDM method in order to achieve a ranking items, and hence an ABC classification and an estimated cost of this classification. The two used methods for our proposed model have been adapted to provide acceptable performance, to comply with the constraints of the addressed problem and to outperform the most common classification algorithms.

To extend this research, it would be interesting to assess the benefits of applying our models empirically using larger datasets. Another avenue for further research consists of hybridization techniques from other different families of methods and compare them with existing models.

References

1. Babai, M., Ladhari, T., Lajili, I.: On the inventory performance of multi-criteria classification methods: empirical investigation. Int. J. Prod. Res. **53**(1), 279–290 (2015)
2. Bhattacharya, A., Sarkar, B., Mukherjee, S.: Distance-based consensus method for ABC analysis. Int. J. Prod. Res. **45**(15), 3405–3420 (2007)
3. Braglia, M., Grassi, A., Montanari, R.: Multi-attribute classification method for spare parts inventory management. J. Qual. Maintenance Eng. **10**(1), 55–65 (2004)
4. Chen, J.: Peer-estimation for multiple criteria ABC inventory classification. Comput. Oper. Res. **38**(12), 1784–1791 (2011)
5. Chen, J.X.: Multiple criteria ABC inventory classification using two virtual items. Int. J. Prod. Res. **50**(6), 1702–1713 (2012)
6. Figueira, J.R., Greco, S., Roy, B., Słowiński, R.: An overview of electre methods and their recent extensions. J. Multi-Criteria Decis. Anal. **20**(1–2), 61–85 (2013)
7. Flores, B., Olson, D., Dorai, V.: Management of multicriteria inventory classification. Math. Comput. Model. **16**(12), 71–82 (1992)
8. Gajpal, P., Ganesh, L., Rajendran, C.: Criticality analysis of spare parts using the analytic hierarchy process. Int. J. Prod. Econ. **35**(1), 293–297 (1994)
9. Güvenir, H.A.: A genetic algorithm for multicriteria inventory classification. In: Artificial Neural Nets and Genetic Algorithms, pp. 6–9. Springer (1995)
10. Guvenir, H.A., Erel, E.: Multicriteria inventory classification using a genetic algorithm. Eur. J. Oper. Res. **105**(1), 29–37 (1998)
11. Hadi-Vencheh, A.: An improvement to multiple criteria ABC inventory classification. Eur. J. Oper. Res. **201**(3), 962–965 (2010)
12. Hadi-Vencheh, A., Mohamadghasemi, A.: A fuzzy AHP-DEA approach for multiple criteria ABC inventory classification. Expert Syst. Appl. **38**(4), 3346–3352 (2011)
13. Kabir, G.: Multiple criteria inventory classification under fuzzy environment. Int. J. Fuzzy Syst. Appl. (IJFSA) **2**(4), 76–92 (2012)
14. Kabir, G., Hasin, M.: Multiple criteria inventory classification using fuzzy analytic hierarchy process. Int. J. Ind. Eng. Comput. **3**(2), 123–132 (2012)

15. Liu, J., Liao, X., Zhao, W., Yang, N.: A classification approach based on the outranking model for multiple criteria ABC analysis. Omega (2015)
16. Mareschal, B., Brans, J., Vincke, P., et al.: PROMETHEE: A new family of outranking methods in multicriteria analysis. ULB-Universite Libre de Bruxelles, Technical report (1984)
17. Mohammaditabar, D., Ghodsypour, S., O'Brien, C.: Inventory control system design by integrating inventory classification and policy selection. Int. J. Prod. Econ. **140**(2), 655–659 (2012)
18. Ng, W.L.: A simple classifier for multiple criteria ABC analysis. Eur. J. Oper. Res. **177**(1), 344–353 (2007)
19. Partovi, F., Burton, J.: Using the analytic hierarchy process for ABC analysis. Int. J. Oper. Prod. Manage. **13**(9), 29–44 (1993)
20. Partovi, F., Hopton, W.: The analytic hierarchy process as applied to two types of inventory problems. Prod. Inventory Manage. J. **35**(1), 13 (1994)
21. Ramanathan, R.: ABC inventory classification with multiple-criteria using weighted linear optimization. Comput. Oper. Res. **33**(3), 695–700 (2006)
22. Roy, B.: ELECTRE III: Un algorithme de classement fondé sur une représentation floue des préférences en présence de critères multiples. Cahiers du CERO **20**(1), 3–24 (1978)
23. Roy, B.: The outranking approach and the foundations of electre methods. Theory Decis. **31**(1), 49–73 (1991)
24. Saaty, T.: The analytical hierarchy process: planning, setting priorities, resource allocation (1980)
25. Storn, R., Price, K.: Differential evolution-a simple and efficient adaptive scheme for global optimization over continuous spaces, vol. 3. ICSI, Berkeley (1995)
26. Storn, R., Price, K.: Differential evolution-a simple and efficient heuristic for global optimization over continuous spaces. J. Glob. Optim. **11**(4), 341–359 (1997)
27. Tsai, C.Y., Yeh, S.W.: A multiple objective particle swarm optimization approach for inventory classification. Int. J. Prod. Econ. **114**(2), 656–666 (2008)
28. Zhou, P., Fan, L.: A note on multi-criteria ABC inventory classification using weighted linear optimization. Eur. J. Oper. Res. **182**(3), 1488–1491 (2007)

A New Hybrid Multi-criteria ABC Inventory Classification Model Based on Differential Evolution and Topsis

Hedi Cherif[1(✉)] and Talel Ladhari[2,3]

[1] Institut Supérieur de Gestion, Université de Tunis, Tunis, Tunisia
cherif.hedi@gmail.com
[2] Ecole supérieure des Sciences Economiques et Commerciales de Tunis Université de Tunis, Tunis, Tunisia
[3] College of Business, Umm Al-Qura University, Umm Al-Qura, Saudi Arabia

Abstract. During the last decades, many companies have taken seriously the task of managing the inventory efficiently because of the surplus of stock and the need to make more profits for their financial and logistical well-being. For this purpose, the ABC classification is one of the most frequently analysis used in production and inventory management domains, in order to classify a set of items in three predefined classes A, B and C, where each class follows a specific management and control policies. In this paper, we present a new hybrid approach for the ABC multi-criteria inventory classification (MCIC) problem using the evolutionary algorithm namely the Differential Evolution (DE) with the multi-criteria decision making method (MCDM), called Topsis. This hybrid approach is modeled by using DE, the parameters of which (criteria weights) are optimized and tuned by using a Topsis method. To evaluate objectively the performance of our proposed model, an estimation function based on the inventory cost and the fill rate service level is used, and also represents the objective function of our approach DE-Topsis, which consists of minimizing the inventory cost. The aim of our proposed approach is to exploit the robustness and usefulness of both DE and Topsis methods, to reduce the inventory cost, to provide acceptable performance and to comply with the constraints of the ABC MCIC problem. A comparative study is conducted to compare our proposed hybrid approach with other ABC classification models of the literature by using a widely used data set. We have established that the proposed model enables more accurate classification of inventory items and better inventory management cost effectiveness for the ABC multi-criteria inventory classification problem.

Keywords: Inventory management · ABC Multi-criteria Inventory Classification · Hybrid model · Differential evolution · Topsis

A. Abraham et al. (eds.), *Proceedings of the 16th International Conference on Hybrid Intelligent Systems (HIS 2016)*, Advances in Intelligent Systems and Computing 552,
DOI 10.1007/978-3-319-52941-7_9

1 Introduction

The inventory classification is a way or more explicitly a strategy in how that inventory will be managed. The most popular and a widely used approach for the inventory classification problem is the ABC analysis, which consists of classifying each inventory item to one group among A, B, or C groups. Each group has an appropriate level of control. This distribution of inventory items into three classes is done according to the annual dollar usage of an inventory item. However, in practice it has been demonstrated that obtaining a good inventory classification is not guaranteed with the traditional one dimensional ABC analysis [12,19]. Thereby, many enterprises have taken seriously the task of replacing the conventional ABC classification by the multi-criteria inventory classification (MCIC), by including others criteria such as ordering cost, criticality, lead time (LT), commonality, obsolescence, substitutability, number of requests, scarcity, durability, substitutability, repairability, order size requirement, stockability, demand distribution, and stock-out penalty cost [5,6,9,13,15,18,19,22,24,29], in order to fit in with constantly and rapid changes.

To deal with MCIC problem, various methodologies have been used in the literature to categorize inventory items by taking into account their computed scores, such as Artificial Intelligence, Multi-Criteria Decision Making and the Mathematical Programming. The MCIC problem was discussed for the first time by Flores and Whybark [9,10], who emphasized the importance to integrate several criteria in order to generate adequate inventory classification, given the global market requirements, and introduce the matrix-based methodology for the multi-criteria ABC classification. The Analytic Hierarchy Process (AHP) method, developed by Saaty [23], is among the most used methods in MCIC. Since, several studies e.g. [3,8,11,20,21] have used the AHP method to solve the MCIC problem in different manners. Cohen and Ernst propose a statistical clustering technique to classify inventory items using a large combinations of attributes [6,7]. Guvenir and Erel [12] have implemented a method that uses the generic algorithm in order to learn criteria weight, and have established cut-off points between the classes A–B and B–C. Tsai and Yeh [27] use the particle swarm optimization technique and present an inventory classification algorithm that simultaneously searches the optimum number of inventory classes and performs classification. Mohammaditabar et al. use the simulating annealing method [17] and developed an integrated model that simultaneously categorizes the items and find the best policy. Yu [28] propose different artificial intelligence-based techniques including support vector machines (SVMs), backpropagation networks, and the k-nearest neighbor (kNN) algorithm in order to classify inventory items. As for Partovi and Anandarajan [19], they applied an artificial neural network (ANN) for ABC classification inventory items, using back propagation and GA as learning methods.

Linear and nonlinear optimization models have been proposed in the literature in order to generate a weight vector that optimizes the weighted score of each item. Ramanathan [22] has developed a weighted linear optimization model (called R model). This R model was further improved by Zhou and Fan [29], who

proposed a ZF-model to address deficiencies and subjectivity of the R model. Ng [18] developed a linear optimization model (called NG model), and Hadi-Vencheh [14] implemented a nonlinear optimization model (called H model), which computes the optimal items score by keeping the weight effects. Chen [4] provides a peer-estimation approach based on two weighted linear optimization models R and ZF, which aggregates two performances scores generated from the R model and the ZF model.

The main contribution of this paper is to combine DE algorithm [25,26] with Topsis method as a new hybrid approach, to tackle the ABC inventory classification problem. In other words, we use the DE method to generate solutions within the constraints of the problem. These solutions represent thereafter the criteria weights and serve as an input parameters for Topsis method, which have the task to generate an items ranking, based on score. An ABC distribution will be subsequently applied to these items to dispatch them between A, B and C classes, in order to obtain an ABC classification. This classification is evaluated by using an inventory cost function, which also represents an objective function of our problem. The aim of the proposed models is especially to reduce the inventory cost compared to existing models in the literature, respecting the standard norms of the ABC classification.

The rest of this paper is organized as follows. In Sect. 2, the used methods of our hybrid approach (DE and Topsis) are briefly presented. The proposed hybrid optimization model is described in Sect. 3. Section 4 presents the dataset, the experimental results and comparative numerical studies with some models from the literature. The paper ends with conclusions and future research.

2 Theoretical Background

2.1 Differential Evolution

The differential evolution method [25,26] is probably one of the most powerful stochastic optimization algorithms. This evolutionary algorithm use NP vectors of D dimensions as the population at each generation G. The steps of DE method are as follows:

- **Initialization:** The vectors of the initial population are randomly generated and cover the entire search space, where each parameter of the initial vector is generated from the following equation:

$$x_{j,i,0} = x_{j,min} + rand_{i,j}[0,1](x_{j,max} - x_{j,min}) \tag{1}$$

  where $x_{j,min}$ and $x_{j,max}$ are respectively the lower and upper bounds of the search space and $rand_{i,j}[0,1]$ is a uniformly distributed random number $\in [0,1]$.
- **Mutation:** A mutant vector $v_{i,G}$ is generated for each target vector $x_{i,G}$, according to:

$$v_{i,G+1} = x_{r1,G} + F(x_{r2,G} - x_{r3,G}) \tag{2}$$

$r1$, $r2$ and $r3$ are randomly chosen indexes, belonging to the set $\{1, 2, ..., NP\}$, mutually different and must be different from the current index i. F represent a real in $\in [0, 2]$ which has an effect on the differential variation amplification $(x_{r2,G} - x_{r3,G})$.

- **Crossover:** This step is useful for diversification of vector parameters. For this, each element of the trial vector $u_{i,G+1}$ is generated as follows:

$$u_{ji,G+1} = \begin{cases} v_{ji,G+1} \text{ if } (rand(j) \leq P_{CR}) \text{ or } j = rnbr(i) \\ x_{ji,G} \quad \text{if } (rand(j) > P_{CR}) \text{ and } j \neq rnbr(i) \end{cases} \tag{3}$$

With $j \in [1, D]$, $rand(j)$ is a random number generated in $[0, 1]$, P_{CR} is the crossover probability $\in [0, 1]$. For ensuring that the trial vector $u_{i,G+1}$ receives at least one parameter from the mutant vector $v_{i,G+1}$, we choose a random index $rnbr(i)$ between 1 and D.

- **Selection:** This step indicates if the target vector will be maintained in the next generation or to be replaced by the trial vector, where $f(x)$ represents the objective function to be minimized.

$$x_{i,G+1} = \begin{cases} u_{i,G} \text{ if } f(u_{i,G}) \leq f(x_{i,G}) \\ x_{i,G} \text{ if } f(u_{i,G}) > f(x_{i,G}) \end{cases} \tag{4}$$

2.2 Topsis

The basic concept of the MCDM method Topsis is that the chosen solution must have the shortest distance to the Positive Ideal Solution (PIS) and the farthest distance from the Negative Ideal Solution (NIS) [2,16]. TOPSIS represents an aggregation function which computes the weighted scores of each alternative and ranks these alternatives, taking into account an immeasurable and conflicting criteria.

To understand the problem, we consider N alternatives A_i $(i = 1, ..., N)$ evaluated by M criteria C_j $(j = 1, ..., M)$. Here the approach of the Topsis method, step by step:

- Construct the performance matrix $X = (x_{ij})_{N,M}$, where each alternative $A_i (i = 1, ..., N)$ is evaluated on the criterion $C_j (j = 1, ..., M)$.
- Obtain the weight of each criterion $w_j (j = 1, ..., M)$, such that:

$$\sum_{j=1}^{M} w_j = 1 \tag{5}$$

- Compute the normalized decision matrix $R = (x_{ij}^n)$:

$$x_{ij}^n = \frac{x_{ij}}{\sqrt{\sum_{k=1}^{N} x_{kj}^2}} \quad j = 1, ..., M \text{ and } i = 1, ..., N. \tag{6}$$

- Compute the normalized weighted decision matrix $V = (v_{ij})_{N,M}$:

$$v_{ij} = w_j x_{ij}^n \quad j = 1, ..., M \text{ and } i = 1, ..., N. \tag{7}$$

- Compute the Positive Ideal Solution (PIS) and Negative Ideal Solution (NIS):

$$PIS = A_i^+ = \{V_1^+, V_2^+, ..., V_m^+\} = \{(max\{v_{ij}\}|j \in B, min\{v_{ij}\}|j \in C)\} \quad (8)$$

$$NIS = A_i^- = \{V_1^-, V_2^-, ..., V_m^-\} = \{(min\{v_{ij}\}|j \in B, max\{v_{ij}\}|j \in C)\} \quad (9)$$

  where B and C represent respectively the sets of benefit and cost criteria and A_i^+ (respectively A_i^-) is the maximum (respectively the minimum) value of v_{ij} among all inventory items.
- Compute the euclidean distance S_i^+ (respectively S_i^-) between each alternative A_i and PIS (respectively NIS):

$$S_i^+ = \sqrt{\sum_{j=1}^{M}(v_{ij} - V_j^+)^2} \; i = 1, ..., N. \quad (10)$$

$$S_i^- = \sqrt{\sum_{j=1}^{M}(v_{ij} - V_j^-)^2} \; i = 1, ..., N. \quad (11)$$

- Compute the score of each alternative A_i:

$$SM_i = \frac{S_i^-}{S_i^+ + S_i^-} \; i = 1, ..., N. \quad (12)$$

3 Application of DE-Topsis to ABC Inventory Classification

Our new proposed hybrid approach DE-Topsis and addressed to the ABC MCIC problem is composed into two steps; the first step is that the DE method generates solutions that respect the constraints of the problem and the second stage starts with the Topsis method that uses these solutions in order to calculate the scores of each item and generate a ranking of these items following the descending order of scores. Finally, an ABC classification is applied to all the items according to a defined class distribution. This classification dispatches all the inventory items into three predefined and ordered classes: the first 20% of items in class A (very important), the next 30% of items in class B (moderately important) and the remaining 50% of items in class C (least important). Each generated ABC classification from each solution will subsequently be evaluated by a classification cost function. All the steps of our new hybrid approach DE-TOPSIS is explained in the following algorithm:

To compare objectively the performance of our proposed approach, an estimation function based on the inventory cost and the fill rate service level is used [1]. By setting the specified service level for each class (0.99 for class A, 0.95 for class B and 0.9 for class C), this inventory performance evaluation method estimates two important measures: the total holding inventory cost for all items and

Algorithm 1. DE-TOPSIS approach

Notation :
s_0 : First initial solution, s : Current solution, s^* : Best solution
$E(s)$: Set of solutions
$DE(s)$: Solution generated by DE method from solution s
$TOPSIS(s)$: Classification generated by TOPSIS method from solution s
f : Objective function, f^* : Best value of the objective function
Initialization :
Generate the initial solutions $s \in E(s)$;
$s^* = s_0$; $f^* = f(TOPSIS(s_0))$;
for $s \in E(s)$ **do**
  if $f(TOPSIS(s)) < f^*$ **then**
    $s^* = s$;
    $f^* = f(TOPSIS(s))$;
  end
end
repeat
  for *each* $s \in E(s)$ **do**
    $s_{DE} = DE(s)$;
    $f(TOPSIS(s))$ = Cost of the solution s
    $f(TOPSIS(s_{DE}))$ = Cost of the solution s_{DE}
    if $f(TOPSIS(s_{DE})) < f(TOPSIS(s))$ **then**
      $s = s_{DE}$;
      if $f(TOPSIS(s)) < f^*$ **then**
        $s^* = s$;
        $f^* = f(TOPSIS(s))$;
      end
    end
  end
until *termination condition is reached*;
return s^*

the achieved overall service level of the system. To present this performance evaluation method, we use the same notations and definitions for the three following formulas as in [1].

The total safety stock inventory cost is given by:

$$C_T = \sum_{i=1}^{N} h_i k_i \sigma_i \sqrt{L_i} \tag{13}$$

The Fill Rate of each item i can be approximated by:

$$FR_i = 1 - \frac{\sigma_i \sqrt{L_i}}{Q_i} G(k_i) \tag{14}$$

The overall FR of the inventory system is calculated as follows:

$$FR_T = \frac{\sum_{i=1}^{N} FR_i D_i}{\sum_{i=1}^{N} D_i} \tag{15}$$

4 Computational Results

The DE method must generate solutions that respect two conditions: (a) the values of the criteria weights generated must be between 0 and 1 and (b) the sum of all weights must be equal to 1. Accordingly, we adapt the DE method to achieve these two conditions, as follows:

$$x_{i,j} = x_{max} - rand\left[0, \left[x_{max} - \sum_{t=1}^{D} x_{i,t}\right]\right] \tag{16}$$

where $x_{max} = 1$ (upper bound of the research space) and D the total of criteria.

The mutation step of the DE method (Eq. 2) generates a mutant vector which can have values outside the lower and upper bounds of the search space (respectively 0 and 1). To address this conflict, we calibrated the values so that the parameters remain within the search space:

$$x_{i,j} = \begin{cases} 0 & \text{if } x_{i,j} < 0 \\ 1 & \text{if } x_{i,j} > 1 \\ x_{i,j} & \text{otherwise.} \end{cases} \tag{17}$$

However, this calibration does not ensure that the sum of the values vector is equal to 1, which represents the sum of criteria weights. For this purpose, we made the vector normalization in order to this sum must be equal to 1, as following:

$$x_{i,j} = \frac{x_{i,j}}{\sum_{t=1}^{D} x_{i,t}} \tag{18}$$

To compare our new hybrid approach with existing literature models in the ABC MCIC problem, we use a data set that contains 47 inventory items evaluated in terms of three criteria and provided by an Hospital Respiratory Therapy Unit (HRTU). This data set displayed in the Table 1 has been widely used in the literature (R model [22], ZF model [29], Chen model [4], H model [13], NG model [18]. Note that all the established classifications respect the same ABC distribution, with 10 items in the class A, 14 items in the class B and 23 items in the class C.

To obtain the best calibration, we have done a factorial experimental desgin with 7*11*8 = 616 different versions of DE-Topsis, where all possible combinations for the DE parameters are tested (Population size: 20, 30, 50, 75, 100, 150, 200; Amplification factor F: 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 1, 1.25, 1.5, 1.75 and

Table 1. Classification of DE-Topsis vs existing models.

Item	ADU	AUC	LT	R [22]	ZF [29]	Chen [4]	H [13]	NG [18]	DE-Topsis
1	5840.64	49.92	2	A	A	A	A	A	C
2	5670	210	5	A	A	A	A	A	A
3	5037.12	23.76	4	A	A	A	A	A	C
4	4769.56	27.73	1	B	C	B	A	A	C
5	3478.8	57.98	3	B	B	B	A	A	C
6	2936.67	31.24	3	C	C	B	B	A	C
7	2820	28.2	3	C	C	B	B	B	C
8	2640	55	4	B	B	B	B	B	B
9	2423.52	73.44	6	A	A	A	A	A	A
10	2407.5	160.5	4	B	A	A	A	A	B
11	1075.2	5.12	2	C	C	C	C	C	C
12	1043.5	20.87	5	B	B	B	B	B	B
13	1038	86.5	7	A	A	A	A	A	A
14	883.2	110.4	5	B	A	B	A	B	A
15	854.4	71.2	3	C	C	C	C	C	C
16	810	45	3	C	C	C	C	C	C
17	703.68	14.66	4	C	C	C	C	C	C
18	594	49.5	6	A	A	B	B	B	A
19	570	47.5	5	B	B	B	B	B	B
20	467.6	58.45	4	C	B	C	C	C	B
21	463.6	24.4	4	C	C	C	C	C	B
22	455	65	4	C	B	C	C	C	B
23	432.5	86.5	4	C	B	C	B	B	B
24	398.4	33.2	3	C	C	C	C	C	C
25	370.5	37.05	1	C	C	C	C	C	C
26	338.4	33.84	3	C	C	C	C	C	C
27	336.12	84.03	1	C	C	C	C	C	C
28	313.6	78.4	6	A	A	A	B	B	A
29	268.68	134.34	7	A	A	A	A	A	A
30	224	56	1	C	C	C	C	C	C
31	216	72	5	B	B	B	B	B	B
32	212.08	53.02	2	C	C	C	C	C	C
33	197.92	49.48	5	B	B	B	B	B	B
34	190.89	7.07	7	A	B	A	B	B	A
35	181.8	60.6	3	C	C	C	C	C	C
36	163.28	40.82	3	C	C	C	C	C	C
37	150	30	5	B	B	B	C	C	B
38	134.8	67.4	3	C	C	C	C	C	C
39	119.2	59.6	5	B	B	B	B	B	B
40	103.36	51.68	6	B	B	B	B	B	A
41	79.2	19.8	2	C	C	C	C	C	C
42	75.4	37.7	2	C	C	C	C	C	C
43	59.78	29.89	5	B	C	C	C	C	B
44	48.3	48.3	3	C	C	C	C	C	C
45	34.4	34.4	7	A	B	A	B	B	A
46	28.8	28.8	3	C	C	C	C	C	C
47	25.38	8.46	5	B	C	C	C	C	B
Classification Cost				927.517	945.357	958.143	999.892	1011.007	**821.444**
Fill Rate				0.986	0.984	0.988	0.99	0.991	0.972

Crossover Probability P_{CR}: 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0). The stop criterion is when a maximum number of generations (1000) or consecutive non-improving generations (100) is reached. The best version of DE-Topsis model requires the following setting: $F = 0.5$, $P_{CR} = 0.5$ and a population of 50 candidates. We clearly observe that our proposed approach provides a more efficient classification cost (821.444) than all other models presented from the literature (927.517 for R-Model [22], 945.357 for ZF-Model [29], 958.143 for Chen-Model [4], 999.892 for H-Model [13] and 1011.007 for NG-Model [18]), with a good Fill Rate (0.972) reflecting a good classification and a customer satisfaction.

5 Conclusion

In this paper, we present a novel hybrid approach addressed to the ABC MCIC problem by combining the Differential Evolution method with the Topsis method. The evolutionary algorithm has been modified in order to adapt it to the requirements of Topsis method and to comply with the constraints of the addressed problem. The aims of our proposed hybrid model DE-Topsis is not solely to classify the inventory items based on objective weights, but especially to reduce the inventory cost, to provide acceptable performance, to comply with the constraints of the problem and to outperform the most common classifications with the same ABC distribution class.

References

1. Babai, M., Ladhari, T., Lajili, I.: On the inventory performance of multi-criteria classification methods: empirical investigation. Int. J. Prod. Res. **53**(1), 279–290 (2015)
2. Behzadian, M., Otaghsara, S.K., Yazdani, M., Ignatius, J.: A state-of the-art survey of topsis applications. Expert Syst. Appl. **39**(17), 13051–13069 (2012)
3. Braglia, M., Grassi, A., Montanari, R.: Multi-attribute classification method for spare parts inventory management. J. Qual. Maintenance Eng. **10**(1), 55–65 (2004)
4. Chen, J.: Peer-estimation for multiple criteria ABC inventory classification. Comput. Oper. Res. **38**(12), 1784–1791 (2011)
5. Chen, Y., Li, K.W., Kilgour, D.M., Hipel, K.W.: A case-based distance model for multiple criteria ABC analysis. Comput. Oper. Res. **35**(3), 776–796 (2008)
6. Cohen, M.A., Ernst, R.: Multi-item classification and generic inventory stock control policies. Prod. Inventory Manag. J. **29**(3), 6–8 (1988)
7. Ernst, R., Cohen, M.: Operations related groups (ORGs): a clustering procedure for production/inventory systems. J. Oper. Manag. **9**(4), 574–598 (1990)
8. Flores, B., Olson, D., Dorai, V.: Management of multicriteria inventory classification. Math. Comput. Model. **16**(12), 71–82 (1992)
9. Flores, B.E., Clay Whybark, D.: Multiple criteria ABC analysis. Int. J. Oper. Prod. Manag. **6**(3), 38–46 (1986)
10. Flores, B.E., Whybark, D.C.: Implementing multiple criteria ABC analysis. J. Oper. Manag. **7**(1), 79–85 (1987)
11. Gajpal, P., Ganesh, L., Rajendran, C.: Criticality analysis of spare parts using the analytic hierarchy process. Int. J. Prod. Econ. **35**(1), 293–297 (1994)

12. Guvenir, H.A., Erel, E.: Multicriteria inventory classification using a genetic algorithm. Eur. J. Oper. Res. **105**(1), 29–37 (1998)
13. Hadi-Vencheh, A.: An improvement to multiple criteria ABC inventory classification. Eur. J. Oper. Res. **201**(3), 962–965 (2010)
14. Hadi-Vencheh, A., Mohamadghasemi, A.: A fuzzy AHP-DEA approach for multiple criteria ABC inventory classification. Expert Syst. Appl. **38**(4), 3346–3352 (2011)
15. Hautaniemi, P., Pirttilä, T.: The choice of replenishment policies in an MRP environment. Int. J. Prod. Econ. **59**(1), 85–92 (1999)
16. Hwang, C., Yoon, K.: Multiple decision attribute making: methods and applications (1981)
17. Mohammaditabar, D., Ghodsypour, S., O'Brien, C.: Inventory control system design by integrating inventory classification and policy selection. Int. J. Prod. Econ. **140**(2), 655–659 (2012)
18. Ng, W.L.: A simple classifier for multiple criteria ABC analysis. Eur. J. Oper. Res. **177**(1), 344–353 (2007)
19. Partovi, F.Y., Anandarajan, M.: Classifying inventory using an artificial neural network approach. Comput. Ind. Eng. **41**(4), 389–404 (2002)
20. Partovi, F., Burton, J.: Using the analytic hierarchy process for ABC analysis. Int. J. Oper. Prod. Manag. **13**(9), 29–44 (1993)
21. Partovi, F., Hopton, W.: The analytic hierarchy process as applied to two types of inventory problems. Prod. Inventory Manag. J. **35**(1), 13 (1994)
22. Ramanathan, R.: ABC inventory classification with multiple-criteria using weighted linear optimization. Comput. Oper. Res. **33**(3), 695–700 (2006)
23. Saaty, T.: The analytical hierarchy process: planning, setting priorities, resource allocation (1980)
24. Stonebraker, P.W., Leong, G.K.: Operations Strategy: Focusing Competitive Excellence. Allyn and Bacon, Boston (1994)
25. Storn, R., Price, K.: Differential Evolution-a Simple and Efficient Adaptive Scheme for Global Optimization over Continuous Spaces, vol. 3. ICSI, Berkeley (1995)
26. Storn, R., Price, K.: Differential evolution-a simple and efficient heuristic for global optimization over continuous spaces. J. Glob. Optim. **11**(4), 341–359 (1997)
27. Tsai, C.Y., Yeh, S.W.: A multiple objective particle swarm optimization approach for inventory classification. Int. J. Prod. Econ. **114**(2), 656–666 (2008)
28. Yu, M.: Multi-criteria ABC analysis using artificial-intelligence-based classification techniques. Expert Syst. Appl. **38**(4), 3416–3421 (2011)
29. Zhou, P., Fan, L.: A note on multi-criteria ABC inventory classification using weighted linear optimization. Eur. J. Oper. Res. **182**(3), 1488–1491 (2007)

Integration of Game Theory in R^2-IBN Framework for Conflict Resolution in a Multi-agents Model of an Extended Enterprise

Alaeddine Dronga[1(✉)], Lobna Hsairi[1,2], and Khaled Ghedira[3]

[1] Higher Institute of Computer Science of Tunis, University of Tunis El Manar, Tunis, Tunisia
dronga.alaeddine@gmail.com
[2] Faculty of Computing and Information Technology, University of Jeddah, Jeddah, Saudi Arabia
hsairil@yahoo.fr
[3] Higher Institute of Management of Tunis, Tunis, Tunisia
khaled.ghedira@isg.rnu.tn

Abstract. This paper aims to improve the argument based negotiation framework R^2-IBN to overcome the cooperation problem within the extended enterprise by the agent-oriented model named MAIS-E^2. In order to overcome the problem of cooperation in general and conflict resolution which prevent this practice in particular, we propose as a solution the integration of game theory foundations into two modules of R^2-IBN framework which are: "Argument Evaluation" module and "Argument selection" module. Finally, we expose the evaluation and the validation of our work through the use of a port application.

Keywords: Multi-agent systems · Game theory-based negotiation · Argumentation-based negotiation · R^2-IBN · Game theory · MAIS-E^2

1 Introduction

These last decades the economic environment in which is situated any enterprise is marked by multiple transformations which do not stop questioning the strategies adopted by enterprises. To face the new constraints of their environment, enterprises had to become allied and grouped together within collective entities by establishing cooperation links. For that purpose, new forms of the enterprises appeared following the extended enterprise example [1].

To face the new stakes, the classic techniques succeed only in proposing limited answers. It is in this context that the agent-oriented approach, branch of the artificial intelligence, emerge as an innovative approach and a promising management for extended enterprises [2]. In fact, the intelligent components (or agents) and their capacities of interaction form a multi-agents system, whose capacities are superior to the simple sum of capacities of the individual agents. Also, they

A. Abraham et al. (eds.), *Proceedings of the 16th International Conference on Hybrid Intelligent Systems (HIS 2016)*, Advances in Intelligent Systems and Computing 552,
DOI 10.1007/978-3-319-52941-7_10

allow having a big adaptability of the systems to their environments, what establishes moreover a fundamental property of the target enterprises as well as the information exchange, the collaboration, the cooperation and the coordination of the actions between the various links in the extended enterprise [3].

Furthermore, when several agents interact, conflict situations can occur, what requires the use of the conflicts resolution mechanisms. In fact, game theory allows to describe and to analyze numerous economic and social relations under the form of strategic games [4]. So, game theory aims to formalize inherent conflict situations from individual community in interaction, to discuss then to propose solutions to these conflicts. Furthermore, it is the branch of the decision theory which concerns the interdependent decisions [5].

For all these reasons, we suggest in this paper to study the integration of game theory foundations in the argument based negotiation framework R^2-IBN (Relationship-Role and Interest Based Negotiation), and this for the objective of improving and optimizing the latter on one hand and to solve the conflicts between the extended enterprise members represented by the agent-oriented model MAIS-E^2 (Multi-Agent Information System for an Extended Enterprise) on the other hand.

The remainder of this paper is structured as follows: Sect. 2 presents related works, Sect. 3 presents multi-agent model for inter-enterprises cooperation MAIS-E^2, Sect. 4 presents the integration of game theory in R^2-IBN, Sect. 5 presents experimental validation of our work. Finally, Sect. 6 concludes the paper and outlines some research directions.

2 Related Works

In the literature little works used game theory foundations in an argument based negotiation such as works of Rahwan and Larson [6] in which they introduced the game theory into an argumentation Mechanism Design (ArgMD) based on abstract reasoning system of Dung [7]. However, the result of negotiation is not only determined by exchanged arguments, but also by strategies used by agents who present these arguments. Thus, agents can be selfish; they can have conflicting preferences on which the arguments will be accepted. For this, they have suggested to apply the tools of game theory in the abstract argumentation framework of Dung.

Furthermore, Hadidi [8] has proposed an argumentation framework for the automated negotiation based on some game theory notions. This framework allows to categorize the offers according to the arguments which possess them and to negotiate by exploiting a much known protocol (Alternating Offers Protocol) used in game theory. The main advantage of this work is the use of this protocol to put a strategy which will be exploited in the preference relations between the offers on one hand and with any form of concession on the other hand.

Besides, Mbarki et al. [9] focus on the strategic aspect in argument based negotiation. They introduce three strategies of concession and four strategies of acceptance. Furthermore, they classify the agents of negotiation in twelve types of

agents on the basis of the adopted strategies. In fact, in each negotiation dialogue, every participant selects either a concession strategy or an acceptance strategy.

In Sect. 3, we present Multi-agent model for inter-enterprises cooperation MAIS-E^2.

3 MAIS-E^2 Model

MAIS-E^2 (Multi-Agent Information System for an Extended Enterprise) model is a model of an extended enterprise cooperation [10]. In fact, the cooperation at the inter-enterprises level is a situation in which several agents interact between them for conflicts resolution problems of which can prevent this cooperation to determine one or several activities in common. Moreover, this cooperation is ensured through the *Mediator* Agent of MAIS-E^2 model.

In Sect. 4, we present Game theory integration in R^2-IBN.

4 Game Theory Integration in R^2-IBN

With the aim to improve the argument based negotiation framework R^2-IBN, we propose to integrate the foundations of game theory within the R^2-IBN framework. This integration will be focused at the "Arguments evaluation" module on one hand and the "Argument selection" module on the other hand. To reach this objective, we use the argument desirability degree as the utility value of each Agent and the arguments of every Agent as the set of the possible strategies.

To this end, our game is defined by a triplet $\langle N, (A_i)_{i \in N}, (U_{ij})_{i \in N, j \in A} \rangle$ with:

- N: group of players (Agents).
- $(A_i)_{i \in N}$: all arguments which can be adopted by the players (Agents).
- $(U_{ij})_{i \in N, j \in A}$: all possible gains.

The strategic form of our game is modeled as shown in Table 1:

Table 1. Example of gain matrix

		Mediator Agent 2			
		A_{21}	A_{22}		A_{2n}
Mediator Agent 1	A_{11}	(0.1, 0.7)	(0.7, 0.2)	...	(1, 0.9)
	A_{12}	(0.3, 0.4)	(0.8, 0.6)	...	(0.6, 0.7)
	⋮	⋮	⋮	...	⋮
	A_{1n}	(0.4, 0.7)	(0.4, 0.7)		(0.8, 0.9)

With $U_{ij} = argument_{ij}$ desirability degree

- i = 1: *Mediator* agent 1
- i = 2: *Mediator* agent 2

– j ∈ [1..n]: indicates the argument number
– The $argument_{ij}$ desirability degree ∈ [0..1]

Obviously, the *Mediator* Agent 1 who sends a locution, possesses a set of arguments as well as the opponent Agent (the *Mediator* 2 Agent). To calculate the values of such matrix, we focus on the work of Hsairi [2] who used fuzzy logic [11] through a fuzzy controller using a Takagi-Sugeno-Kang system (TSK) [12]. The latter, used to calculate the argument desirability degree that we consider as the utility value of each agent.

4.1 Mediator Agent Reasoning Cycle

In this section, we introduce and explain *Mediator* Agent reasoning cycle in a negotiation situation in the R^2-IBN framework after the integration of game theory foundations, the main contribution of our work. This reasoning cycle is illustrated by Fig. 1.

Before presenting this cycle, it is interesting to precise that the evaluation of exchanged arguments between the two agents will be made through game theory integration that allows us to deduce if compromise can be reached or not. As beginning, the *Mediator* Agent proceeds to evaluate the proposal as well as the argument already grafted to the received locution. This evaluation

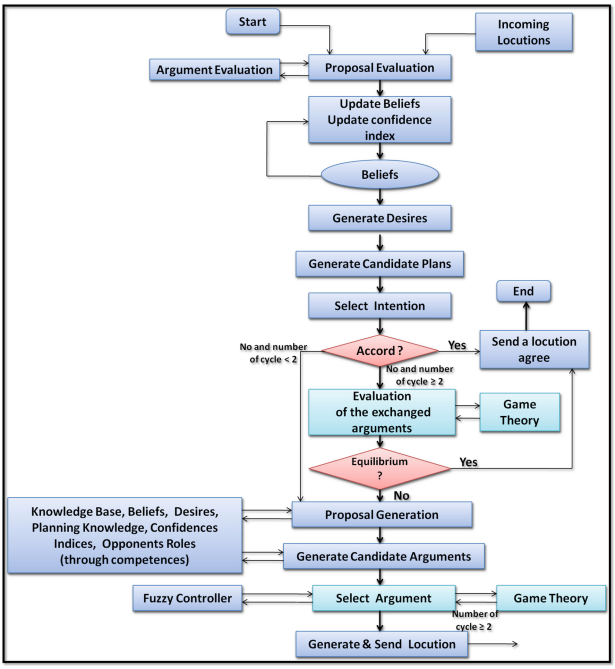


Fig. 1. Reasoning cycle of the Mediator Agent

can involve adapting agent's beliefs as well as opponent Agent confidence index, which induces new beliefs that have consequences new desires generation and candidate plans for its completion. Plan selection among the candidate plans produces the intention agent. If the agent intention is to solve the conflict, the agent sends a compromise locution.

In the other hand (when a compromise is not reached), two cases can be distinguished:

– A compromise is not reached with at least two negotiation cycles[1].
– A compromise is not reached with a number of negotiation cycle strictly less than two.

In this regard, we explain both cases previously enumerated as follows.

First case (a compromise is not reached with at least two negotiation cycles)
The *Mediator* Agent evaluates all the exchanged arguments based on the foundations of game theory to see if it can reach a compromise when using the game theory approach. More exactly, the *Mediator* Agent models the exchanged arguments with the opponent Agent in the strategic form game. After this step, the agent tries to find at least one *Nash* equilibrium. If the *Mediator* Agent finds at less an equilibrium (Whatever it is a pure strategy equilibrium or a correlated equilibrium) then the *Mediator* Agent sends an agreement locution.

Otherwise the *Mediator* Agent generates a proposal from its knowledge base, its desires and planning knowledge, etc. As well as generating a set of candidate arguments that can be grafted to the generated proposal.

At this step of the algorithm, a question which arises: what is the best argument to be grafted in the proposal from the point of view of two agents ? To answer this question, we apply the fundamentals of game theory to select an argument[2].

If the *Mediator* Agent finds an equilibrium, the argument which constitutes the equilibrium will be grafted in the proposal to be sent in order to improve the chance that this argument will convince the opponent Agent and therefore reach a compromise. Let us note that we consider the arguments already received because we can consider them as an unveiling of the beliefs of the opponent Agent.

Otherwise, if there is no game equilibrium, then the *Mediator* Agent sends the argument judged as the best for him through a fuzzy controller (without the use of game theory).

[1] Note that a negotiation cycle means that agent sends a proposal and the opponent Agent answers it.

[2] Note that in "Argument selection" module the strategic form game represents the candidates arguments of the *Mediator* Agent which will be grafted in the proposal to be sent to the opponent Agent on the one hand and the arguments received on the other hand.

Second case (a compromise is not reached with a number of negotiation cycle strictly less than two)
The *Mediator* Agent generates a proposal from its knowledge base, its desires and planning knowledge, etc. As well as generating a set of candidate arguments that can be grafted to the generated proposal. A question arises: what is the best argument to be grafted in the proposal? As a response, we propose the activation of a fuzzy controller for the selection of this argument.

The final phase for the two cases listed above, is to generate and send a locution to the opponent Agent. Let us note that this reasoning cycle of the *Mediator* Agent will be repeated until reaches a compromise and that argument will be sent only once.

To summarize, we propose the integration of game theory within the R^2-IBN framework. More specifically, we integrated the game theory in the "Arguments evaluation" module and in the "Argument Selection" module.

In the next section, we present the various application steps of game theory in the "Arguments evaluation" module.

4.2 Application Steps of Game Theory in the "Arguments Evaluation" Module

To apply the game theory in the "Arguments evaluation" module, several steps have been taken as shown in Fig. 2. First, the *Mediator* Agent defines the different arguments in order to model them in a normal form game. Then the *Mediator* Agent seeks a *Nash* equilibrium of pure strategy in this matrix.

- In the case of a single *Nash* equilibrium, then a compromise is reached wherein the *Mediator* Agent grafts the corresponding argument to the *Nash* equilibrium in the proposal to send.
- In the case of several *Nash* equilibriums, the *Mediator* Agent seeks among these equilibriums if there is an optimal equilibrium by applying the Pareto-optimal theorem. If the answer is positive (that is to say, is there an optimal Pareto equilibrium) then the *Mediator* Agent sends the found argument through this equilibrium to the opponent Agent and a compromise is reached. Otherwise, the *Mediator* Agent selects the *Nash* equilibrium that maximizes its gain and a compromise is reached and end of the negotiation.
- Otherwise, the *Mediator* Agent seeks a correlated equilibrium[3]. For the latter, two cases can be presented:
  - If the expected gains of each *Mediator* Agent for the equilibrium are found in the matrix, then the *Mediator* Agent sends the argument which constitutes the equilibrium. In this case a compromise is reached.
  - Otherwise (that is to say, the correlated equilibrium outcome is not found in the matrix), the *Mediator* Agent passes to the "Proposal Generation" module and follow the remaining steps in the reasoning cycle of *Mediator* Agent and so on until reaching a compromise.

[3] The correlated equilibrium is a generalization of the concept of *Nash* equilibrium in mixed strategies.

The Fig. 2 summarizes this whole process:

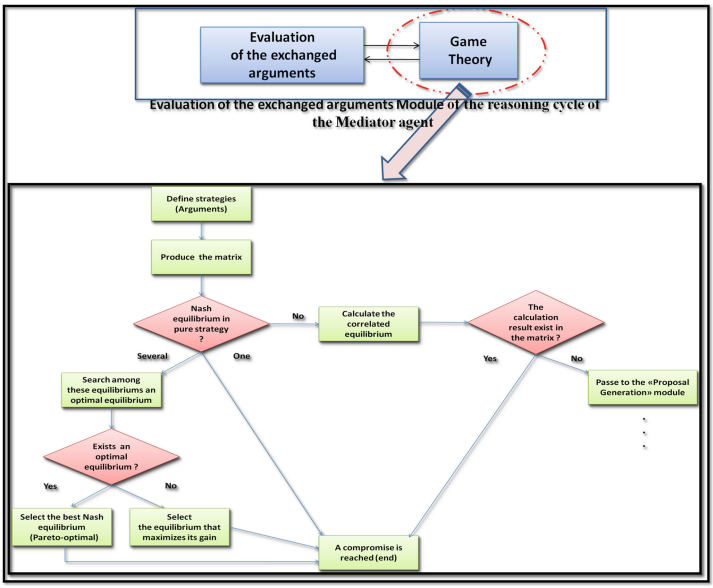


Fig. 2. Operational steps of the "Evaluation of the exchanged arguments" module

After the explanation of the different steps of application of game theory in the "Arguments evaluation" module, it is propitious to provide the application steps of the game theory in the "Argument Selection" module.

4.3 Application Steps of Game Theory in the "Argument Selection" Module

In order to integrate game theory foundations in the "argument selection" module, several steps have been taken as shown in Fig. 3. First, the *Mediator* Agent defines the different arguments in order to model them in a normal form game. Then, the *Mediator* Agent seeks a *Nash* equilibrium of pure strategy in this matrix.

- In the case of a single *Nash* equilibrium, the *Mediator* Agent grafts the corresponding argument to the *Nash* equilibrium in the proposal to send.
- In the case of several *Nash* equilibriums, the *Mediator* Agent seeks among these equilibriums if there is an optimal equilibrium by applying the Pareto-optimal theorem. If the answer is positive (that is to say, is there an optimal equilibrium) then the *Mediator* Agent sends the found argument through this equilibrium to the opponent Agent. Otherwise, the *Mediator* Agent selects the *Nash* equilibrium that maximizes its gain and the argument constituting the selected equilibrium will be sent to the opponent Agent.

– Otherwise, the *Mediator* Agent seeks a correlated equilibrium. For this equilibrium, two possibilities can be presented:
  - If the expected gains of each *Mediator* Agent for the equilibrium are found in the matrix, then the *Mediator* Agent sends the argument which constitutes this equilibrium.
  - Otherwise, the *Mediator* Agent sends the argument judged best for him by applying the fuzzy controller that allows to perform this operation (that is to say without the integration of game theory).

The Fig. 3 summarizes this whole process.

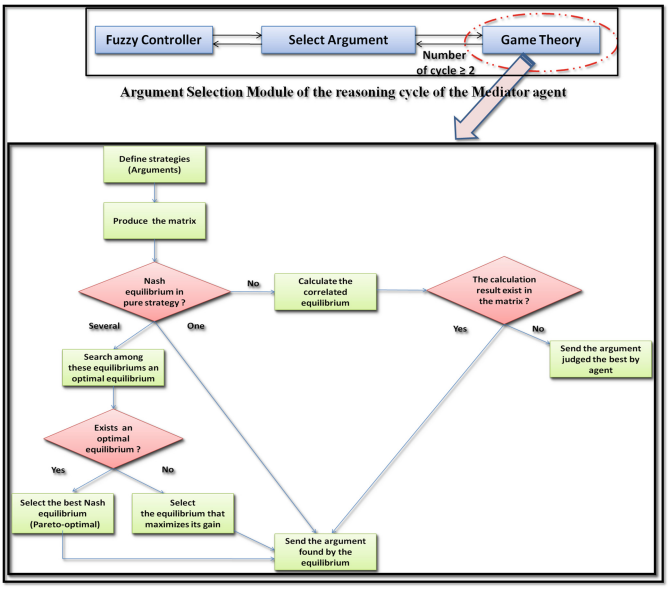


Fig. 3. Operational steps of the "Argument Selection" module

5 Experimental Validation

To validate our work and answer mentioned questions (conflict resolution/ argumentation-based negotiation/game theory based negotiation), we used a port application, considered as an example of an extended enterprise and this through the JADE multi-agent platform.

Results and comparative study

Tables 2 and 3 summarize our experimental results. Indeed, Tables 2 and 3 show the experimental results of ten examples of negotiation performed between the

Table 2. Experimental results of the argumentation based negotiation (R^2-IBN)

Example No.	First price offered by the Ship-owner Agent	First price offered by the Maritime Agent	Accepted price	Number of negotiation cycles	Number of exchanged arguments
1	5300	4800	5100	10	17
2	5400	5000	5200	12	21
3	6000	5300	5650	16	29
4	4900	4600	4850	11	19
5	4000	3500	3650	14	25
6	3500	3300	3450	9	15
7	3000	2400	2750	10	17
8	2600	2000	2150	11	19
9	2100	1600	2000	13	23
10	1500	1000	1200	8	13

Table 3. Experimental results of the integration of game theory in R^2-IBN

Example No.	First price offered by the Ship Owner Agent	First price offered by the Maritime Agent	Accepted price	Number of negotiation cycles	Number of exchanged arguments
1	5300	4800	5200	6	9
2	5400	5000	5100	10	17
3	6000	5300	5550	13	23
4	4900	4600	4800	8	13
5	4000	3500	3700	11	19
6	3500	3300	3450	9	15
7	3000	2400	2750	10	17
8	2600	2000	2550	9	15
9	2100	1600	1950	12	21
10	1500	1000	1200	8	13

Maritime Agent and the *ship-owner* Agent for the price to be paid by the *Maritime* Agent to the *ship-owner* Agent for the transport of goods. Obviously, our goal is to resolve conflicts between these two agents.

For this, the Table 2 shows the experimental results of the argumentation based negotiation only (R^2-IBN) more clear without the integration of game theory while Table 3 shows the experimental results of game theory based negotiation and the argumentation based negotiation. More specifically, the integration of game theory in R^2-IBN framework.

In order to analyze these two tables, with the aim to perform a comparative study of these different results, we can observe that the integration of game theory in the R^2-IBN framework (as shown in Table 3) is better in terms of number of negotiation cycle and therefore the number of exchanged arguments that an argumentation based negotiation (R^2-IBN) as shown in Table 2.

Therefore, the obtained results show that the number of negotiation cycle and the number of exchanged arguments are less or equal than the case of integration of game theory when searching for convergence towards agreement.

6 Conclusion

In this paper, we handled the problem of cooperation in the extended enterprise. Our main goal was to improve and optimize the argumentation based negotiation framework R^2-IBN. For this, we have proposed the integration of the game theory foundations in the argumentation based negotiation framework R^2-IBN as a solution allowing this type of enterprise to overcome the problems of cooperation in general and conflicts resolution that hindered this practice in particular.

In fact, we have shown the various steps of the integration of game theory in two modules of this framework: "Arguments Evaluation" module and the "Argument Selection" module.

The works conducted under this paper open the way to many perspectives and further works, of which we can mention:

- The integration and application of game theory fundamentals in R^2-IBN for several agents (n agents).
- The study of the interaction complexity between agents in the case of cardinality extension of the negotiation through an empirical analysis.

References

1. Villarreal Lizárraga, C.L., Dupont, L., Gourc, D., Pingaud, H.: Contributing to management of shared projects in SMEs clusters. In: Proceedings of the 18th International Conference on Production Research, Salerno, Italy (2005)
2. Hsairi, L., Ghédira, K., Alimi, A.M., Ben Abdellhafid, A.: MAIS-E^2 model and R^2-IBN framework: port application case. In: Industrial and Logistics Systems, (SIL 2010), Marrakech (2010)
3. Bellaaj, S.: Adaptation to the risks of outsourcing strategies: Tunisian industrial companies case. Ph.D. thesis, University of Sfax (2009)
4. Rasmusen, E.: Games and Information: An Introduction to Game Theory, 4th edn. Wiley-Blackwell, Oxford (2006). 582 p.
5. Hans, P.: Game Theory a Multi-Leveled Approach, 2nd edn. Springer, Berlin (2015)
6. Rahwan, I., Larson, K.: Argumentation and Game Theory. In: Argumentation in Artificial Intelligence, pp. 321–339. Springer (2008)
7. Dung, P.M.: On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. Artif. Intell. **77**, 321–357 (1995)

8. Hadidi, N.: Unification of Argumentation and Game Theory for Automated Negotiation. Ph.D. thesis, University Paris Descartes (Paris 5) (2012)
9. Mbarki, M., Marey, O., Sultan, K.: Agent types and adaptive negotiation strategies in argumentation-based negotiation. In: 26th International Conference on Tools with Artificial Intelligence (ICTAI), pp. 485–492 (2014)
10. Hsairi, L., Ghédira, K., Alimi, A.M., Benabdelhafid, A.: Resolution of conflicts via argument based negotiation: Extended enterprise case. In: IEEE International Conference on Services Systems and Services Management (IEEE/SSSM06), pp. 828–833. Université de Technologie de Troyes, France (2006)
11. Zadeh, L.A.: Is there a need for fuzzy logic? Inf. Sci. **178**(13), 2751–2779 (2008)
12. Sugeno, M., Kang, G.: Structure identification of fuzzy models. Fuzzy Sets Syst. **28**, 15–33 (1988)

A Novel Approach of Deep Convolutional Neural Networks for Sketch Recognition

Lamyaa Sadouk(✉), Taoufiq Gadi, and El Hassan Essoufi

Faculty of Science and Technology Settat, Settat, Morocco
lamyaa.sadouk@gmail.com, gtaoufiq@yahoo.fr,
e.h.essoufi@gmail.com

Abstract. Deep Neural Networks (DNNs) have recently achieved impressive performance for many recognition tasks across different disciplines including image recognition task. However, most of existing works on deep learning for image recognition focus on natural image data (photo-based images) and not on sketches. Moreover, most of existing works on sketch classification are based on hand crafted feature representations. In this paper, we propose to train a convolutional neural network for sketch recognition using the TU-Berlin sketch dataset composed of 250 object categories with 80 images each. We find that training a CNN with a proper data-augmentation and a multi-scale multi-angle voting technique can achieve an accuracy of 75.43%, which surpasses human-level performance in the standard sketch classification benchmark and significantly outperforms state-of-the-art sketch recognition methods.

Keywords: Sketch · Recognition · Learning · Deep learning · Convolutional neural network

1 Introduction

Sketches are known to be simple and hasty drawings, giving the essential features without the details. Even though they are considered to be a rough design of the reality, they are an effective communicative tool for humans. And, with the growing field of touchscreens, sketching became more and more popular. Since then, sketching became an attractive field of research, giving rise to many applications including sketch recognition [1, 6], sketch-based image retrieval [13, 16], sketch-based 3D model retrieval [26], and forensic sketch analysis [28, 29].

Sketches and natural photo-based images are very different in appearance even though they appear to have a lot of things in common and convey the same meaning. Compared to natural images, sketches are very *abstract* and *don't have color or texture*. Consequently, existing methods for natural images cannot be directly applied to our sketch recognition problem. Moreover, the same object can be drawn with different levels of detail (abstraction), e.g., every person has his or her own way to sketch an object, drawing either a stickman with a rough structure or a portrait with fine details. Therefore, *internal structures of sketches are very complex and details of the drawing are very sparse*. All these reasons make the sketch classification a more challenging task. In fact, even humans can only achieve a recognition accuracy of 73.1% [1].

A. Abraham et al. (eds.), *Proceedings of the 16th International Conference on Hybrid Intelligent Systems (HIS 2016)*, Advances in Intelligent Systems and Computing 552,
DOI 10.1007/978-3-319-52941-7_11

Previous sketch recognition works extract hand-crafted features then feed them to a classifier. Some works use both binary shape descriptors, such as chamfer matching [35] and shape context [5], while other works use natural image descriptors, such as SIFT [26] and HOG [12]. The natural descriptor SIFT has shown to do quite well on sketch recognition problems since it captures strong gradient information. Nevertheless, it is only able to describe local information in sketch, which makes SIFT more suitable for natural image (photo-based) recognition rather than sketch recognition.

In this paper, we propose to learn sketch features for sketch classification using a deep learning method, i.e. convolutional neural networks (CNN). Recently, CNNs have shown a great success in pattern recognition since features are not hand-crafted but are learned directly and automatically from images. CNNs are very efficient in extracting all level features (low-, middle- and high-level features), which is exactly what we need to detect all levels of abstraction contained in sketch images. In our work, we apply CNN on sketches the same way they were applied to images, keeping in mind that natural image recognition and sketch recognition are two different fields –as mentioned below.

Our contribution is to get a high sketch classification accuracy even with a limited number of training data ($\sim$13500 training images with 56 images per category), by: (i) training a CNN with a novel configuration and, (ii) using a new strategy during testing phase that is based on multi-angle voting.

In the sections below, we review related works of sketch recognition and deep neural networks (Sect. 2), we then describe our training CNN model as well as our testing method (Sect. 3). Then, in Sect. 4, our experiments are carried out and our sketch classification results are compared to state-of-art methods. Finally, Sect. 5 summarizes our work.

2 Related Work

2.1 Natural Photo-Based Image Classification

Many works on local features have been investigated for the natural image domain, including textons [17], histogram of oriented gradients (SIFT [20] and HOG [4]), bag of visual words [3, 27], sparse [34] and local coding, super vector coding [37], VLAD [10] and Fisher Vectors [23]. Recently, deep neural networks, especially convolutional ones [15, 25, 35] have shown a great success in natural image recognition, dominating top benchmark results on visual recognition tasks. Unlike conventional local feature representations that use shallow handcrafted features, convolutional neural networks is an end-to-end deep model trained from image to class label, containing millions of parameters that are learned from data.

2.2 Sketch Recognition Using Hand-Crafted Feature Representation

Sketch recognition didn't get much attention until 2012 when a large crowd-sourced dataset was published in [1]. This dataset contains 20,000 unique sketches evenly distributed over 250 object categories, which are totally completed by no-expert free hands (examples of hand-sketched objects are shown in Fig. 1). The dataset was used

Fig. 1. Examples of hand-sketched objects

in [1] to perform a sketch recognition study on both the humans and the SIFT descriptor method. Since then, this dataset was used for future works on sketch recognition [5, 6] applying handcrafted features representation borrowed from photos and using the SVM classifier. Moreover, Li et al. [17] found that fusing different local features using multiple kernel learning produces better recognition accuracy. Also, by applying each feature individually, they showed that HOG feature achieves a better accuracy than others. Lately, the work of [21] introduced Fisher Vectors as a new method for sketch recognition. All of the works mentioned above use hand-crafted local natural image features for sketch recognition.

Even though many of recognition applications apply low-level descriptors design [23, 25, 36, 42, 44, 45], most of the sketch recognition works do not make use of sketch shape features to design low-level descriptors that are specific to sketches. They one take into consideration the low-level features used on natural images, since they outperform shape features. Their methods are simple and undergo three easy steps: (a) selecting the most informational areas or points; (b) computing the best local features; (c) building a structure model as well as an evaluation scheme by spatial cues (structural information) or local feature similarity (matching score). These frameworks seriously depend on parameters of the chosen local features and models selected. This implies that it is necessary to play randomly with combinations of features and models until the best recognition score is achieved. Moreover, because of the abstract aspect of sketches, these latter can be hardly classified with models using specific local features. That explains why none of the above methods was able to exceed the humans' performance in sketch recognition.

2.3 Sketch Recognition Using Automated Feature Representation

Deep learning has been proven to be very effective for various image recognition tasks, e.g., image classification, semantic segmentation, image retrieval, shape classification, etc. However, existing works. In particular, Convolutional Neural Networks (CNNs) have dominated top benchmark results on visual recognition challenges such as ILSVRC [7]. When first introduced in the 1980s, CNNs were the preferable solution for small problems only (e.g. LeNet [14] for handwritten digit recognition). They didn't gain popularity since they were restricted by the high computational cost due to the

large network architecture and training data. It was not until the emergence of GPUs, ReLU activation neurons (instead of TanH), max-pooling (instead of average pooling) and dropout regularisation [15] that CNNs achieved very high performance.

CNN has been extended to many traditional areas such as face recognition [19], object detection [18], and image retrieval [21]. And most of them have achieved better results than traditional methods. Unlike conventional hand-crafted features, CNN is an end to-end model trained from image to class label.

Sketch has always been treated as low-level concepts such as shape, edge map, contour or strokes. Since a total sketch is associated with a real object or scene, this makes the sketch a high-level concept, which is far more difficult than natural images. Recently, there are some methods using CNN for sketch recognition, such as [2, 31, 39, 40]. Later, works of [2, 39] use the TU Berlin dataset of Eitz et al. [1] and build their own CNNs that outperforms the human performance, achieving an accuracy of 74.9% and 75.42% respectively. Meanwhile, [31] trains two popular CNNs – Imagenet CNN and a modified version of LeNet CNN, but it doesn't use the whole dataset of Tu Berlin Benchmark. Instead, it eliminates drawn sketches whose identity is difficult to discern for human beings and is left with a subset containing 160 non-ambiguous object categories with 56 sketches each. And very recently came the work [40] which trains a pre-trained Alexnet model using an edge preserving technique and a multi-angle voting at test time, performing 77.3% accuracy.

Our paper uses the original TU Berlin too. So our results will be compared to the performance of [2, 39, 40]. Our paper proposes to: (i) train a new CCN based on a novel data augmentation technique, (ii) perform the testing phase using a new approach based on a multi-angle and multi-scale voting.

3 Our Work - Methodology

In this section, we introduce the key components of our approach. First, we show how dataset images were preprocessed and how our data was augmented (Sect. 3.1), knowing we need to feed the network with lots of data in order for the neural network to be properly trained. Next, we discuss the network architecture that will be applied for the training phase (Sect. 3.2). Then, for the testing phase, we will explore our multi-angle and multi-scale voting strategy (Sect. 3.3). Figure 2 illustrates our overall framework.

3.1 Preprocessing Dataset Images and Data Augmentation

Before training our CNN, it is necessary to preprocess input data images. Next, the training dataset needs to be increased by replicating images and undergoing transformations on them.

Preprocessing Images: Each image in the dataset is 1111 by 1111 pixels. Since images are relatively big in size, they cannot be fed directly to the network. They need to be resized and shrunk to a fixed size before training. Images are then resized to 224×224.

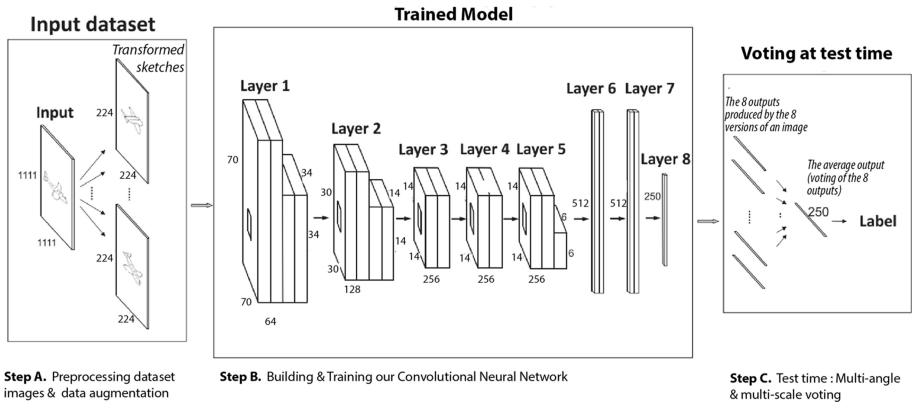


Fig. 2. Illustration of the overall framework

Data Augmentation: Generally, deep neural networks perform very well on image recognition provided that enough training examples are fed to them. Otherwise, with a small number of training examples, the network tends to overfit, providing wrong results during testing. To avoid overfitting, the training dataset should be large [1, 13, 22]. But, in our case, training dataset is very small (56 images per category). One way to enlarge our dataset is to use data augmentation.

In our work, data augmentation consists of replicating images with several transformations. In each epoch, we select half of the batch images then perform the following transformations for each of the input image. We do: (i) a horizontal reflection i.e. mirroring (across vertical axis), (ii) a random rotation between $-30°$ and $30°$, (iii) a rescaling of the width and the height of the image independently using a random resizing range i between 0.4 and 1, (iv) and a horizontal translation to the right with a random range between 0 to 224 - *image size/i.* Later in the experiment section, we will show that the proposed resizing method achieves better sketch recognition accuracy.

3.2 Building Our Own CNN Model from Scratch

Our network is a deep convolutional neural network that can be viewed as the combination of two elements: a training criterion and a model (e.g., a family of functions, a parameterization) on the one hand, and on the other hand, a particular procedure for approximately optimizing this criterion [33]. So, we need to configure hyper-parameters associated with the optimizer-typically the function class, regularizer and loss function, as well as hyper-parameters associated with the model itself.

Hyper-Parameters of the Approximate Optimization

In our work, we used the stochastic (mini-batch) gradient descent as our gradient-based optimization method. Therefore, the hyper-parameters that need to be tuned are as follows: learning rate, momentum, mini-batch and number of training iterations.

Annealing the Learning Rate: One way to optimize hyper-parameter of our convolutional neural network is to decrease or adapt the learning rate schedule. Choosing the proper learning rate and schedule (i.e. changing the value of the learning rate as learning progresses) can be fairly difficult though.

The first method used is the standard one which consists of using a small enough constant learning rate 0.001 that gives stable convergence over all epochs (full pass through the training set). This methods works well in practice but its training is very slow.

The second method, which is the one we used, is to start with a learning rate of 0.01 and anneal it at each iteration as the convergence slows down, as suggested in Eq. (1) by the work of Bergstra and Bengio [41]:

$$\alpha_{epoch} = \frac{\alpha_0 \tau}{\max(epoch, \tau)} \tag{1}$$

which keeps the learning rate constant for the first τ steps and then decreases as epochs are iterated.

An adaptive and heuristic way of automatically setting τ above is to keep α_{epoch} constant until the training criterion stops decreasing significantly (by more than some relative improvement threshold) from epoch to epoch.

Momentum: The momentum used is a fixed value of 0.9.

Mini-Batch Size: The mini-batch size is typically chosen between 1 and a few hundreds. Our batch size is chosen to be 135. The impact of the batch size on the results is mostly computational, i.e., larger batch size yield faster computation but requires visiting more examples in order to reach the same error.

Number of Training Iterations: The number of epochs is 300.

Hyper-Parameters of the Model & Training Criterion
Hyper-parameters that are related to the "model" and "criterion" are as follows: the number and size of hidden layers as well the regularizer.

Number and Size of Hidden Layers: Each layer in a multi-layer neural network typically has a size that we are free to set and that controls capacity. Because of dropout –a regularizer that will be mentioned in the next section of the paper, it is mostly important to choose the number of layers large enough. Generally, there is no general rule for designing the architecture of CNNs. However, recent CNNs [8, 9] agree on the following architecture: several convolutional layers with ReLU and MaxPooling followed by fully connected layers.

Our specific architecture is as follows: first we use three convolutional layers, each with rectifier (ReLU) [30] units and max pooling (Maxpool).

Our CNN has eight layers, each having a different configuration as show in in Table 1. Unlike the existing pre-trained models applied on natural images, the filter size of the first convolutional layer is chosen to be relatively large 15×15 – compared to 11×11 filter in VGG-F model [8] and 3×3 filter in VGG-deep-16 model [9]. Indeed, sketch images have texture information i.e. small shapes. Therefore, as mentioned in [2],

Table 1. Architecture of our model from scratch

Id	Layer	Type	Filter size	Filter num	Stride	Pad	Output size
0		Input	–				224 × 224
1	L1	Conv	15 × 15	64	3	0	70 × 70
2		ReLu	–	–	–	–	70 × 70
3		Maxpool	3 × 3	–	2	0	34 × 34
4	L2	Conv	5 × 5	128	1	0	30 × 30
5		ReLu	–	–	–	–	30 × 30
6		Maxpool	3 × 3	–	2	0	14 × 14
7	L3	Conv	3 × 3	256	1	1	14 × 14
8		ReLu	–	–	–	–	14 × 14
9	L4	Conv	3 × 3	256	1	1	14 × 14
10		ReLu	–	–	–	–	14 × 14
11	L5	Conv	3 × 3	256	1	1	14 × 14
12		ReLu	–	–	–	–	14 × 14
13		Maxpool	3 × 3	–	2	0	6 × 6
14	L6	Conv (FC)	6 × 6	512	1	0	1 × 1
15		ReLu	–	–	–	–	1 × 1
16		Dropout	–	–	–	–	1 × 1
17	L7	Conv (FC)	1 × 1	512	1	0	1 × 1
18		ReLu	–	–	–	–	1 × 1
19		Dropout	–	–	–	–	1 × 1
20	L8	Conv (FC)	1 × 1	250	1	0	1 × 1

using small filters is not necessary since larger filters are able to capture more structured context rather than textured information.

The filter size of the fourth convolutional layer is 5 × 5, which is the same as the output feature map from previous pooling layer, making it a fully connected layer. Then we have two more fully connected layers. Dropout regularization is applied on the first two fully connected layers. The final layer has 250 output units corresponding to 250 categories (that is the number of unique classes in the TU-Berlin sketch dataset), upon which we place a Softmax loss.

Regularizer: At this point, expanding the net by inserting an extra fully-connected doesn't help much. Also, running similar experiments with fully-connected layers containing 1,000 neurons instead of 500 doesn't yield a better classification result. This means that the network is overfitting. We could then use stronger regularization techniques as an alternative to reduce the tendency to overfit. One common regularizer is the use of dropout regularization technique (weights of the network cannot be updated, nor affect the learning of the other network nodes). With dropout, the learned weights of the nodes become somewhat more insensitive to the weights of the other nodes and learn to decide somewhat more by their own (and less dependent on the other nodes they're connected to). In other words, dropout removes individual activations at random while training the network, which makes the model more robust to

the loss of individual pieces of evidence, and less likely to rely on particular idiosyncrasies of the training data.

In our case, since the sketch dataset is very small (about 56 images per category), we set the dropout rate to a high value 0.5.

Neuron Non-linearity: In our experiments, we used the ReLu as the non-linearity hyper-parameter.

3.3 Multi-angle and Multi-scale Voting During Test Time

After training our model, it is time to go to the testing phase. In the conventional way, we simply feed each testing image into the trained network then see whether the computed output corresponds to the true label of the image. However, in our paper, for each and every testing image, we get multiple transformed versions of that image (rotated and rescaled variants) to vote for the output, which is then compared to the true label of that image.

The idea is to take advantage of the characteristics of our trained model. As we know, our model was trained on an augmented data composed of original images as well as multiple rotated and rescaled variants of the original images. This implies that our network will have no trouble recognizing rotated and rescaled images during testing. Our goal is to take advantage of this characteristic in our network during the testing phase by testing each image several times in different ways:

i. We perform multiple transformations to the image including 4 rescaled sizes (*197, 167, 141* and *119*) and 2 angle rotations (*−25°, 25°*), as illustrated in Table 2. We then obtain 8 variants of the image;
ii. Each one of these 8 images is fed into our network to produce 8 outputs;
iii. The 8 outputs from the 8 versions of the image are averaged before making a prediction, as shown in Eq. (2). For a given image input, the average output $S(c_i)$ of a class label i is given by:

$$S(c_i) = \frac{1}{n}\sum_{j=1}^{n} r_j(c_i) \tag{2}$$

where $r_j(c_i)$ is the output of the image variant j (rotated or rescaled variant of the original image) of the class label i. In our case, we have 8 variants per image, so $n = 8$.

Consequently, the output i.e. the final prediction is a result of the voting of all transformed versions of one image;

iv. We pick the maximum output class label to be the predicted class label, as illustrated in Eq. (3). Then, we compare it with the true label of the image.

$$C = argmax_i S(c_i) \tag{3}$$

Table 2. Results of our trained model at test time using different characteristics. For each image, we perform rotation and rescaling to get n variants of the image. Then, these images are fed into our network, producing n outputs. After that, we see which output has the highest number of votes. Finally, this output is compared to the true label of the image

Angle variant(s)	Size variant(s)	Accuracy at test time
$0°$	224×224	0.717
$-25°\ 0° + 25°$	224×224	0.731
$-25°\ 0° + 25°$	$224 \times 224{,}168 \times 168$	0.746
$-25°\ 0° + 25°$	$224 \times 224{,}168 \times 168,\ 127 \times 127$	0.753
$-25°\ 0° + 25°$	$224 \times 224{,}197 \times 197,\ 167 \times 167,\ 141 \times 141,\ 119 \times 119$	0.7543

We follow these steps for every image in the testing dataset to get the final accuracy.

Our approach is similar to networks ensemble method where multiple networks in the ensemble are trained independently by backpropagation and where their corresponding outputs (i.e. predictors) are averaged at test time to get the final prediction. In practice, the network ensemble method is known to be a useful approach to improve the performance of neural networks. While this method averages the predictions produced by different models for one single image as input, our method combines the predictions, i.e. outputs from only one trained model of multiple versions of an image.

Our approach was also inspired by the work of [40] in which multi-angle voting is performed on computed Softmax output probabilities. However, our work uses multi-angle and multi-scale voting directly on outputs rather than Softmax output probabilities because it appears to give a better performance.

4 Experiments and Results

We perform our sketch recognition evaluation on the TU Berlin dataset. Our experiments include classification and retrieval. TU Berlin [1] has 20,000 non-expert sketches, which are drawn by humans, divided into 250 categories. Each category has 80 sketches and each sketch size is 1111 by 1111. According to [1]'s method of Third cross validation, we randomly divide the dataset into three equal parts, and choose two parts as training data and the remaining part as testing/validation data. We take the average accuracy of the validation set as the final evaluation.

After preprocessing the image and resizing it to 224 by 224, we perform data augmentation by rotating, flipping, resizing and translating the image (as discussed previously).

After training our model–explained in Sect. 3.2, we achieve a classification accuracy of *0.717* (Table 2).

After combining the outputs of only 2 rotated variants ($-25° + 25°$) of each image, the accuracy goes up to *0.731*.

Additionally, if we use the outputs of 2 rotated variants ($-25° + 25°$) and of 1 rescaled variant (the 168 by 168 rescaled image of the 224 by 224 original image), we get a higher accuracy of *0.746*. We notice that more predictions (coming from transformed versions of the image) tend to contribute to a higher accuracy.

Going further by adding more predictions with 2 rotated variants ($-25° + 25°$) and 2 rescaled variants (the 168 by 168 rescaled image and the 127 by 127 rescaled image) produces a slightly higher accuracy *0.753*.

Nonetheless, adding more predictions with 2 rotated variants and 4 rescaled variants helps improve the accuracy by only 0.0013. We deduce that, as the number of transformed versions of the image in the ensemble increases, the performance typically monotonically improves though with diminishing returns.

Training a deep network yields good classification results because of the chosen architecture. Indeed, large filters (15 by 15) used in the first layer of our model catch more details than first layer filters of traditional natural-image recognition models (in which the size vary from 3 by 3 to 11 by 11). As we can see in Fig. 3, most of the first layer filters of our model have strong edges and look like Gabor filters.

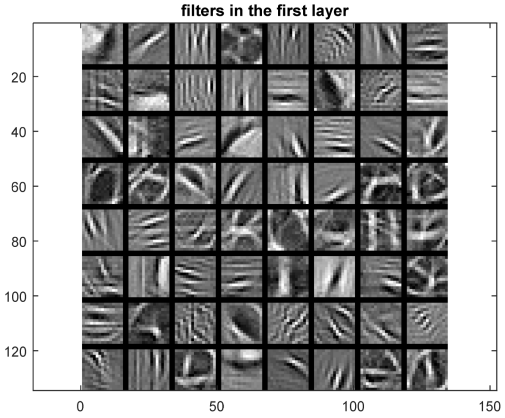


Fig. 3. The first layer output of our CNN model

Comparative Results: Our results as well as multiple results of previous sketch recognition works are shown in Table 3. We can make the following remarks:

(1) *Comparison between our model and works based on hand-crafted features:* Our model without multi-angle and mutli-scaling voting produces an accuracy of 0.717, which is higher than all hand-crafted feature works including HOG-SVM [1], Stargraph + KNN [22], MKL-SVM [5] and FV-SP-SVM [6].

(2) *Comparison between our model and works based on automated features:* Our model using multi-angle and mutli-scaling voting performs better than Sketch-a-net [2] and slightly better than DeepSketch work [39]. This shows that our model can easily compete with state-of-the-art deep learning recognition works.

Table 3. Comparison of different sketch classification methods as well as the human re cognition performance

Method	Accuracy
Humans' recognition [1]	0.732
HOG-SVM [1]	0.56
Stargraph + KNN [22]	0.615
MKL-SVM [5]	0.658
FV-SP-SVM [6]	0.689
Sketch-a-Net [2]	0.749
DeepSketch [39]	0.7542
Deep sketch feature for cross domain image retrieval [40]	0.773
Our model	0.7543

However, our model has 0.0187 less accuracy than Xinggang Wang et al. (may 2016) model. This latter fine-tunes and trains Alexnet, a pre-trained CNN model using the ImageNet image classification data, which is originally designed for 1000 categories; it has 5 convolutional layers and 3 fully connected layers. For the input layer, the model is only applicable to 3 by 227 by 227 input images.

Even though [40] achieves a higher accuracy than ours, our model happens to be simpler because:

(i) Our model uses only one channel per input image whereas the model in [40] uses three channels per input image. So, to use the model in [40], it is necessary to replicate the one channel sketch input images into 3 channel input images, which takes longer processing time and larger memory than our model.
(ii) Our model architecture has 6.8e + 06 parameters i.e. 26 MB of memory, while the model architecture in [40] has 5.8e + 07 parameters i.e. 220 MB of memory; which makes our model less complex, computationally less expensive and less time consuming.
(iii) At test time, our model uses 2 rotated and 4 rescaled variants of the image for voting the final prediction, that is 6 transformed variants in total. However, [40] uses 18 rotated angle variants.

Running Cost: Our models were implemented using the MatConvNet [1] toolbox of Matlab. We trained our model from scratch for 500 epochs, with each epoch having different input data thanks to random data augmentation. The training was completed in approximately 68 h on a 2.60 GHz CPU (without explicit parallelisation). At test time, our model requires less than 5 min to compute the accuracy.

5 Conclusion

In this paper, we proposed a study of deep neural network sketch recognition methods. First, we employed random data augmentation by doing unlimited random transforms to the TU Berlin dataset. After that, we trained our model from using a deep

architecture. At test time, we applied the multi-angle and multi-scale voting technique to get a good classification accuracy of 0.7534. Then, we compared our work with recent works and showed that method is able to outperform most of the up-to-date works. From this sketch recognition study, we conclude that it is possible to increase the performance of image recognition deep networks in general by (i) augmenting training data using a number of transformations on data images i.e. flipping-rotation-rescaling-translation, (ii) applying the same kind of transformations on test images to get several variants per test image, (iii) feeding these variants into the network and applying the voting technique on the predicted outputs. We deduce that increasing the training dataset with geometric transformations makes the network more trained and more capable of recognizing distorted sketch images. Indeed, due to the free-hand nature, the same object can be drawn in different ways and with varied levels of detail/abstraction; so rotating and rescaling image can help us cope with these issues. We believe that adding more types of transformation on the training and testing data will help improve the classification results.

References

1. Eitz, M., Hays, J., Alexa, M.: How do humans sketch objects? In: Proceedings of the SIGGRAPH 2012 (2012)
2. Yu, Q., Yang, Y., Song, Y.-Z., Xiang, T., Hospedales, T.M.: Sketch-a-net that beats humans. In: Proceedings of the British Machine Vision Conference (BMVC), pp. 1–12 (2015)
3. Csurka, G., Dance, C.R., Dan, L., Willamowski, J., Bray, C.: Visual categorization with bags of keypoints. In: Proceedings of the ECCV Workshop on Statistical Learning in Computer Vision (2004)
4. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: CVPR (2005)
5. Li, Y., Hospedales, T.M., Song, Y., Gong, S.: Free-hand sketch recognition by multikernel feature learning. In: CVIU (2015)
6. Schneider, R.G., Tuytelaars, T.: Sketch classification and classification-driven analysis using fisher vectors. In: SIGGRAPH Asia (2014)
7. Deng, J., Dong, W., Socher, R., Li, L., Li, K., Fei-Fei, L.: Imagenet: a large-scale hierarchical image database. In: CVPR (2009)
8. Chatfield, K., Simonyan, K., Vedaldi, A., Zisserman, A.: Return of the devil in the details: delving deep into convolutional nets. In: BMVC (2014)
9. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. In: ICLR (2015)
10. Jégou, H., Douze, M., Schmid, C., Pérez, P.: Aggregating local descriptors into a compact image representation. In: CVPR (2010)
11. Hinton, G.E., Srivastava, N., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.R.: Improving neural networks by preventing co-adaptation of feature detectors. arXiv:1207.0580 (2012)
12. MindFinder: Interactive Sketch-Based Image Search on Millions of Images (2010)
13. Eitz, M., Hildebrand, K., Boubekeur, T., Alexa, M.: Sketch-based image retrieval: benchmark and bag-of-features descriptors. TVCG **17**(11), 1624–1636 (2011)
14. Le Cun, Y., Boser, B., Denker, J.S., Henderson, D., Howard, R.E., Hubbard, W., Jackel, L. D.: Handwritten digit recognition with a back-propagation network. In: NIPS (1990)

15. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: NIPS (2012)
16. Hu, R., Collomosse, J.: A performance evaluation of gradient field hog descriptor for sketch based image retrieval. CVIU **117**(7), 790–806 (2013)
17. Leung, T., Malik, J.: Representing and recognizing the visual appearance of materials using three-dimensional textons. IJCV **43**(1), 1 (2001)
18. Girshick, R., Donahue, J., Darrell, T., Malik, J.: Rich feature hierarchies for accurate object detection and semantic segmentation. In: 2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 580–587. IEEE (2014)
19. Taigman, Y., Yang, M., Ranzato, M., Wolf, L.: Deepface: closing the gap to humanlevel performance in face verification. In: 2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Columbus, Ohio, pp. 1701–1708. IEEE (2014)
20. Lowe, D.G.: Distinctive image features from scale-invariant key-points. IJCV **2**(60), 91–110 (2004)
21. Babenko, A., Slesarev, A., Chigorin, A., Lempitsky, V.: Neural codes for image retrieval. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014. LNCS, vol. 8689, pp. 584–599. Springer, Heidelberg (2014). doi:10.1007/978-3-319-10590-1_38
22. Li, Y., Song, Y.-Z., Gong, S.: Sketch recognition by ensemble matching of structured features. In: British Machine Vision Conference (BMVC). Citeseer (2013)
23. Perronnin, F., Dance, C.: Fisher kernels on visual vocabularies for image categorizaton. In: CVPR (2006)
24. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M.S., Berg, A.C., Fei-Fei, L.: Imagenet large scale visual recognition challenge. CoRR, abs/1409.0575 (2014)
25. Sermanet, P., Eigen, D., Zhang, X., Mathieu, M., Fergus, R., Le-Cun, Y.: Overfeat: integrated recognition, localization and detection using convolutional networks. CoRR, volume abs/1312.6229 (2014)
26. Wang, F., Kang, L., Li, Y.: Sketch-based 3D shape retrieval using convolutional neural networks. arXiv preprint arXiv:1504.03504 (2015)
27. Sivic, J., Zisserman, A.: Video Google: a text retrieval approach to object matching in videos. In: ICCV (2003). 1
28. Klare, B.F., Li, Z., Jain, A.K.: Matching forensic sketches to mug shot photos. TPAMI **33** (3), 639–646 (2011)
29. Ouyang, S., Hospedales, T., Song, Y., Li, X.: Cross-modal face matching: beyong viewed sketches. In: ACCV (2014)
30. LeCun, Y., Bottou, L., Orr, G.B., Müller, K.-R.: Efficient BackProp. In: Orr, Genevieve, B., Müller, K.-R. (eds.). LNCS, vol. 1524, pp. 9–50Springer, Heidelberg (1998). doi:10.1007/3-540-49430-8_2
31. Sarvadevabhatla, R.K., Babu, R.V.: Freehand sketch recognition using deep features. arXiv:1502.00254 (2015)
32. Vedaldi, A., Lenc, K.: MatConvNet: CNNs for MATLAB (2014). http://www.vlfeat.org/matconvnet/
33. Bengio, Y.: Practical Recommendations for Gradient-Based Training of Deep Architectures. arXiv:1206.5533 (2012)
34. Yang, J., Yu, K., Huang, T.: Supervised translation-invariant sparse coding. In: CVPR (2010)
35. Zeiler, M.D., Fergus, R.: Visualizing and understanding convolutional networks. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014. LNCS, vol. 8689, pp. 818–833. Springer, Heidelberg (2014). doi:10.1007/978-3-319-10590-1_53

36. Zhou, B., Lapedriza, A., Xiao, J., Torralba, A., Oliva, A.: Learning deep features for scene recognition using places database. In: NIPS, pp. 487–495 (2014)
37. Zhou, X., Yu, K., Zhang, T., Huang, Thomas, S.: Image classification using super-vector coding of local image descriptors. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010. LNCS, vol. 6315, pp. 141–154. Springer, Heidelberg (2010). doi:10.1007/978-3-642-15555-0_11
38. Frazão, X., Alexandre, L.A.: Weighted convolutional neural network ensemble. In: Bayro-Corrochano, E., Hancock, E. (eds.) CIARP 2014. LNCS, vol. 8827, pp. 674–681. Springer, Heidelberg (2014). doi:10.1007/978-3-319-12568-8_82
39. Seddati, O., Dupont, S., Mahmoudi, S.: DeepSketch: deep convolutional neural networks for sketch recognition and similarity search. In: 13th International Workshop on Content-Based Multimedia Indexing (CBMI), June 2015
40. Wang, X., Duan, X., Bai, X.: Deep sketch feature for cross-domain image retrieval. Elsevier (2016)
41. Bergstra, J., Bengio, Y.: Random search for hyper-parameter optimization. J. Mach. Learn. Res. **13**(2012), 281–305 (2012)

Hybrid TDNN-SVM Algorithm for Online Arabic Handwriting Recognition

Ramzi Zouari[1(✉)], Houcine Boubaker[1], and Monji Kherallah[2]

[1] National School of Engineers of Sfax (ENIS), University of Sfax, Sfax, Tunisia
ramzi.zouari@gmail.com, houcine-boubaker@ieee.org
[2] Faculty of Sciences of Sfax, University of Sfax, Sfax, Tunisia
monji.kherallah@enis.rnu.tn

Abstract. This paper deals with a new system of online Arabic handwriting recognition based on the association of beta-elliptic modeling extractor with a hybrid Time Delay Neural Network (TDNN) and Support Vector Machines (SVM) classifier. The beta-elliptic model proceeds by a segmentation of the handwriting trajectory into fragments called Beta strokes by inspecting the extremums points of the curvilinear velocity and extracting their corresponding static and dynamic profile proprieties. These features are used to train the Time Delay Neural Network which is able to represent the sequential aspect of the input data. The fuzzy outputs of this network are then used to train SVM in order to predict the correct label class. To evaluate our method, we have used a total of 25000 Arabic letters from the LMCA database. Experimental results demonstrate the effectiveness of our proposed method and show recognition rate reaching the 99.52%.

Keywords: Time delay neural network · Kernel · Beta-elliptic model · Beta impulse · Elliptic arc · Receptive fields · Shared weights

1 Introduction

Arabic handwriting recognition is a challenging task compared to other languages because of the cursive style of the handwriting. It contains 28 basic letters and the majority of them change their shape according to their position in the word (beginning, middle, end or isolated) [1]. Furthermore, the handwriting style change from one person to another and depends on the age and emotional state of the writer.

Handwriting processing differs depending on the form of acquisition of the input signal: online or offline. In the first case, the input data is obtained through a scanned image of the handwriting, in this case only static information's are available. In the second case, dynamic information's are available such as (x, y) coordinates, velocity profile, pen pressure and temporal order of trajectory [2]. In the literature, several systems are based on holistic approaches where handwriting analysis is made without segmentation. Among these approaches, we can cite the system of Maalej et al. [3], who presents an online system based on deep recurrent neural networks (RNN) with a Long Short Term Memory. This network is trained by a sequence of handwriting trajectory coordinates (x, y). To protect their network against the over fitting they used the Dropout

A. Abraham et al. (eds.), *Proceedings of the 16th International Conference on Hybrid Intelligent Systems (HIS 2016)*, Advances in Intelligent Systems and Computing 552,
DOI 10.1007/978-3-319-52941-7_12

technique which also improved the recognition rate to being 99.7%. Abdelazeem and Hesham [4] extracted the local and vicinity features of Tunisian town names. These features are trained with a continuous Hidden Markov Model (HMM) with lexicon reduction and tested on ADAB database. The recognition rate is 92.5% in top1 and reached 96.03% in top10. Nakkach et al. [5] propose a method for online handwriting recognition which uses the (x, y) coordinates to calculate the Fourier descriptor and the chain code. The proposed method is tested used the SVM (Support Vector Machine) classifier and achieved a recognition rate of 92.43%. The systems mentioned above used features extracted from a whole shape of the handwriting which can be easily obtained in off-line handwriting. However, some researches use the analytical approaches where each character/word is segmented into sub-units treated independently.

A study was made by Kherallah et al. [6] has adapted an approach that segment the trajectory according to the extremum velocity profile. These segments are converted into visual coding based on freeman method. In this sequence of visual codes, it applied the basic operations of genetic algorithm (GA) such as mutation and cross-over. This approach is assessed on LMCA database and showed an identification rate of 97%. Charfi et al. [7] used the same technique of segmentation developed in [6] and proposed a recognition system based on Graph-matching. This approach is tested on Postal Addresses extracted from ADAB database and obtained a recognition rate about 98%. Daifallah and Jamous [8] uses another segmentation method based on arbitrary segmentation followed by segmentation enhancement, consecutive joints connection and finally segmentation point locating. The handwriting segments are represented by their trajectory points coordinate are modeled through HMM which reach recognition rate up to 97%. In addition to the segmentation methods, several researches proposed the hybridization of multiple classifiers and the combination of online and off-line features. In this context, Tagougui et al. [9] proposed a hybrid MLP/HMM modeling approach. The outputs of the MLP are decoded using a Discrete HMM with a codebook of size 512. This architecture is tested on ADAB database and achieved recognition rate about 96.45%. Hamdani et al. [10] combine multiple HMMs using both online and offline features. This system is evaluated using the IFN/ENIT database and improved the recognition accuracy from 63.8% to 81.93%. Elleuch et al. [11] combined online and offline features using a deep classifier called Convolution Deep Belief Network (CDBN) and they obtained recognition rate up to 97% on LMCA database.

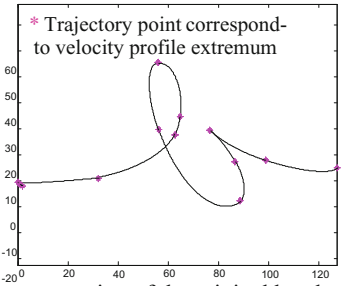
We present in this paper a new method of online Arabic handwriting recognition based on the association of the Beta-elliptic model and the Time Delay Neural Network. This association which represents the contribution of this study, aims to appropriate the sequentially time dependent description delivered by the Beta-elliptic model stage to the successively time delayed entry of the features data at the input of the TDNN classifier stage. In addition, our proposed system is the first to use a TDNN in Arabic handwriting treatment and also the first to associate the beta-elliptic descriptor used in many fields on online handwriting [12] with TDNN. This paper is organized as follows: in the second section we present the beta-elliptic model. In Sect. 3, we give a description of the TDNN architecture and the applied training algorithm. Section 4 presents our proposed system. In the Sect. 5 we present the experimental results before ending by an analytical discussion and presentation of future work.

2 The Beta-Elliptic Model

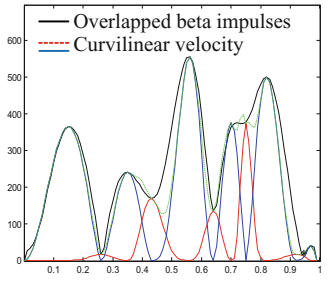
The beta-elliptic approach allows combining two profiles of writing action: velocity and trajectory [13]. The curvilinear velocity Vσ(t) is given by the following formula:

$$V_\sigma(t) = \sqrt{\left(\frac{dx}{dt}\right)^2 + \left(\frac{dy}{dt}\right)^2} \tag{1}$$

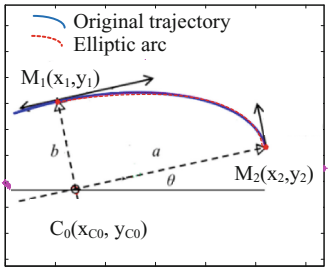
The curvilinear velocity of a continuous cursive handwriting shows a signal profile starting from a zero speed, then alternate minimum and maximum speed and finish by an almost zero speed. The occurrence of velocity extremums correspond also to extrema of variation of the trajectory tangent deviation angle. The handwriting trajectory is then segmented into simple movements called Beta strokes delimited by the occurrences correspondent to local maximums, minimums and double inflexion points of the curvilinear velocity signal [14] (see Fig. 1).



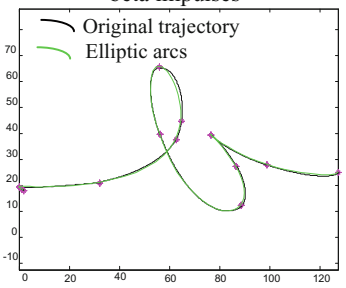
(a). segmentation of the original handwriting of an Arabic middle 'هـ' character



(b). Dynamicprofile modeling by overlapped beta impulses



(c). Trajectory stroke fitting by elliptic arc



(d). geometric profile modeling by elliptic arc

Fig. 1. Dynamic and geometric modeling of a handwritten middle 'Ha' trajectory

2.1 Velocity Profile Modeling

Plamondon and Alimi [15] has proved that curvilinear velocity of one handwriting stroke can be approximated by a Beta impulse. Therefore the curvilinear velocity is the

result of an algebraic addition of the overlapped Beta impulses corresponding to its successive segmented strokes (see Fig. 1-b).

$$V_\sigma(t) \approx \sum_{i=1}^{n} K_i \times \beta_i(t, q_i, p_i, t_{0i}, t_{1i}) = V_r(t) \tag{2}$$

$$\beta_i(t_i, q_i, p_i, t_{0i}, t_{1i}) = \begin{cases} \left(\frac{t-t_{0i}}{t_{Ci}-t_{0i}}\right)^{p_i} \cdot \left(\frac{t_{1i}-t}{t_{1i}-t_{Ci}}\right)^{q_i} & \text{if } t \in [t_{0i}\,,\, t_{1i}] \\ 0 & \text{elsewhere} \end{cases} \tag{3}$$

With

- K_i: amplitude of the i^{th} beta function
- p, q are intermediate parameters
- t_{0i}, t_{ci}, t_{1i} are the moments witch correspond respectively to the start, the maximum amplitude and the end of the Beta function ($t_{0i} < t_{ci} < t_{1i}$)

$$t_{Ci} = \frac{(p \times t_{1i}) + (q \times t_{0i})}{p+q} \tag{4}$$

2.2 Geometric Profile Modeling

In the space domain, each stroke delimited between two consecutive points of velocity extremum is modeled by an elliptic arc defined by the dimensions of its long and small axes a, b and its inclination angle θ (see Fig. 1-c) calculated using its two endpoints position $M_1(x_1, y_1)$ and $M_2(x_2, y_2)$ and their respective trajectory tangents deviation angles $\theta_1$and θ_2 [16]. The following figures present the results of dynamic and geometric profiles fitting of an Arabic on-line handwritten character by respectively overlapped Beta impulses and elliptic arcs.

3 Time Delay Neural Network

The TDNN, Time Delay Neural Network is a convolution neural network (CNN) which was first introduced for speech recognition [17]. The architecture of this network is derived from a Multi-Layer Perceptron (MLP) with back-propagation learning. In MLP, the order of input features doesn't affect the result of training because each neuron in hidden layer is fully connected to all neurons of the input layer simultaneously (global view), whereas in TDNN each neuron in hidden layer is connected to a subset of neurons from the preceding layer that represent a receptive field. Therefore TDNN is very useful for a sequential data [18]. TDNN incorporate a very important notion which is weight sharing. Then the neurons in different receptive fields shared the same weights. This constraint reduces the number of parameters in the system and facilities the generalization process. To build a TDNN we have to define the number of layers, the size of receptive fields that must be the same between two

consecutive layers, and finally the value of temporal shift between receptive fields called Delay. The architecture of the TDNN consists of two principal parts (see Fig. 2). The first present the lower layers and implements the convolutions. It permits to transform the input features into another sequence of higher order features. The second part is similar to MLP and contains the outputs of network.

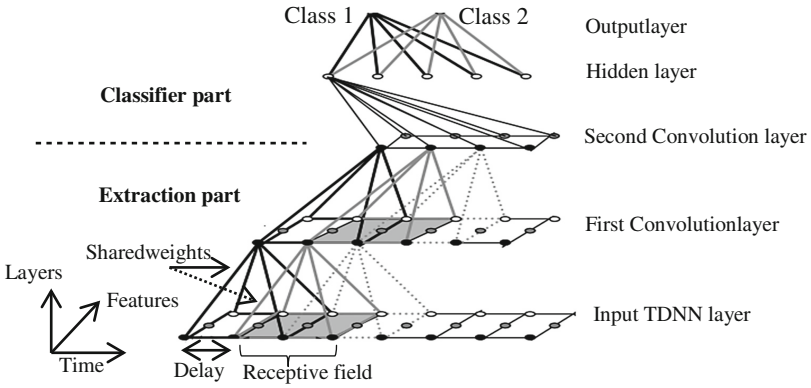


Fig. 2. TDNN architecture

This architecture was used by Poisson and Lallican [19] and was tested on IRONOFF and UNIPEN databases and obtained significantly higher performances near to 98%. In other hand, TDNN allow to reduce the number of coefficients from 36,110 with MLP to 17,930 with TDNN.

4 Proposed System

This part presents the different steps of our recognition system. First, we describe the preprocessing stage, then the feature extraction module and finally we detailed the recognition engine with Pure TDNN and hybrid TDNN-SVM algorithms.

4.1 Preprocessing

Before feature extraction, some processes on online handwriting are necessary. In our case all handwriting shapes are normalized in size and we applied a Chebyshev second type low pass filter on the normalized trajectory to eliminate the noise introduced by temporal and spatial sampling [20].

4.2 Features Extraction

In this step, we extract the dynamic and geometric features of normalized handwriting. This is made by a beta-elliptic model as described in Sect. 2. For each stroke, we

extract 4 dynamic parameters (K, Δt, Rap and P) resulting from the beta modeling of the trajectory kinematic profile and 3 other geometric parameters (a, b and θ) representing the elliptic arc modeling the trajectory static profile. The resulting feature vector is then composed of 7 features explained in the following table [21] (Table 1):

Table 1. Component of beta-elliptic vector features

Features	Parameters	Explanation
Dynamic profile	K	Beta impulse amplitude
	$\Delta t = (t1 - t0)$	Beta impulse duration
	$Rap = \frac{\Delta t_c}{\Delta t} = \frac{p}{p+q}$	Rapport of Beta impulse asymmetry or culminating time
	P	Beta shape parameters
Geometric profile	a	Ellipse major axis half length
	b	Ellipse small axis half length
	θ	Ellipse major axis inclination angle

4.3 Recognition Engine

The proposed recognition system is based essentially on Time delay neural network (TDNN). The choice of this algorithm is justified by its ability to represent the sequential input data thanks to the principle of the receptive field (Fig. 3).

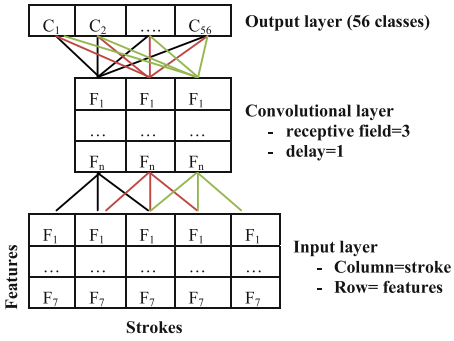


Fig. 3. Explanation of Pure TDNN recognition engine

Pure TDNN Recognizer Engine. The proposed recognition system is based essentially on Time delay neural network (TDNN). The choice of this algorithm is justified by its ability to represent the sequential input data thanks to the principle of the receptive field.

$$nb_{Conv} = \frac{nb_{strokes} - size_{receptive-field}}{delay} + 1 \tag{5}$$

The input layer consists in a 2D matrix where the number of columns depends on the number of strokes and the rows represent the feature vectors (7 features per column). Then each receptive field is fully connected to a set of neurons in the convolution layer which are defined empirically ($F_1 \rightarrow F_n$). At last, the output layer contains all membership classes ($C_1 \rightarrow C_{56}$).

Hybrid TDNN-SVM Recognizer Engine. Support Vector Machines (SVM) is a power classifier developed by Vapnik in 1995 [22] that performs classification tasks. It allows constructing hyper planes in a multidimensional space that separates different class labels. The hyper planes are made by using a mathematical functions, known as kernels such as linear, polynomial, radial basis function (RBF) and sigmoid. Among these kernels, RBF perform better in nonlinear separation case [23] which is defined as follow:

$$K(x, x') = \phi(x).\phi(x') = \exp\left(-\gamma \|x - x'\|^2\right) \tag{6}$$

Where γ: parameter will be defined empirically, x: input vector, Ø: nonlinear transform and Ø(x): transformed features space.

In our hybrid recognizer engine, SVM aims to control the LogSoftMax output layer which returns the membership probabilities to the different classes. In fact, the training step allows SVM to establish a relation between the fuzzy TDNN outputs and the desired outputs of the training samples. Thus the second stage which is composed by SVM has to correct the classification result decision obtained by the LogSoftMax output layer (see Fig. 4).

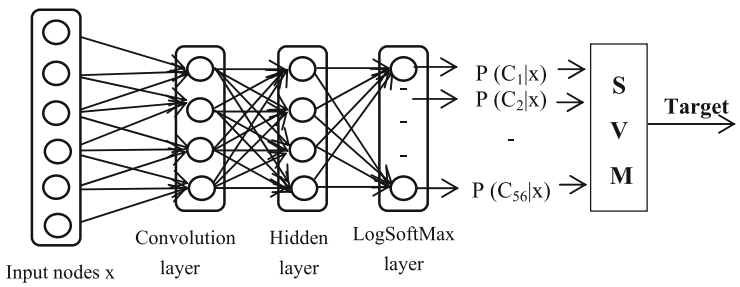
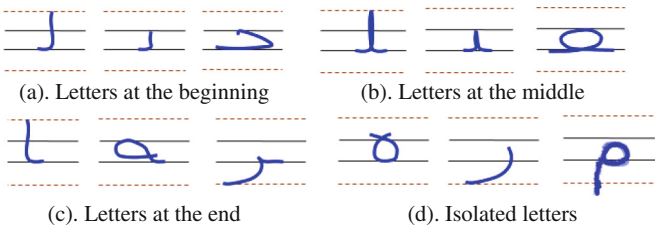


Fig. 4. Explanation of the used hybrid TDNN-SVM recognizer engine

5 Experiments, Results and Discussion

5.1 Experiments and Results

To test and assess our proposed system, we have used the LMCA (Letter Mot Chiffres Arabes) database for our experiments [24]. LMCA contain the different letters of Arabic language in their different positions in the word (beginning, middle, end or isolated). It is composed of 212 files containing the (x, y) coordinates of 23141 handwriting letters belonging to 56 classes. The delayed strokes and diacritics are ignored. The following figure shows some letters from LMCA database [16] (Fig. 5).



(a). Letters at the beginning (b). Letters at the middle

(c). Letters at the end (d). Isolated letters

Fig. 5. Letters from LMCA database

We divided LMCA into two parts (training and test). Two thirds of letters has been used in training phase and the rest has been used in recognition test (see Table 2). Experiments are made using three architectures: Pure TDNN, SVM and hybrid TDNN-SVM. In these architectures, beta-elliptic parameters are calculated and entered into the input layer.

Table 2. Repartition of LMCA database

	Training stage	Test stage
Number of files	140	72
Total number of characters	15427	7714
Number of characters/class	275	138

In both TDNN and hybrid TDNN-SVM, we fixed the learning rate and momentum values at 0.0001 and 0.6 respectively. The size of receptive field and temporal delay are defined empirically. The following table shows the different results of tests were made on Pure TDNN with several numbers of layers and neurons along feature axis (Table 3).

Table 3. Results of Pure TDNN test experiments (receptive field = 3, delay = 1)

Number of layers	Neurons along feature axis	Epochs	Recognition rate	Time (mn)
1 conv. + 1 hidden	30	40	91.89%	5
1 conv. + 2 hidden	40	40	94.1%	9
2 conv. + 1 hidden	40–100	60	**95.16%**	12

The second test is performed with SVM using different kernel functions. SVM with RBF kernel reached a recognition rate slightly better than TDNN but more slower in execution time (see Table 4).

Table 4. Results of SVM classifier tests with beta elliptic parameters

	Linear	Sigmoid	RBF
Time (mn)	9	15	20
Recognition rate	86.16%	92.52%	**96.90%**

The last test consists on pairing TDNN with SVM. The beta-elliptic parameters are entered into input layer. Thereafter, the fuzzy outputs of LogSoftMax layer are used to train SVM with RBF kernel. The same principle is reproduced in test stage (see Fig. 4). This architecture achieves an impressive rate of 99.52% (Table 5).

Table 5. Results of hybrid TDNN-SVM test experiments

Number of layers	Neurons along feature axis	Epochs	Recognition rate	Time (mn)
1 conv. + 1 hidden	30	40	97.40%	24
1 conv. + 2 hidden	40	40	97.92%	29
2 conv. + 1 hidden	40–150	50	**99.52%**	27

5.2 Discussion

Through various tests, we concluded that TDNN don't require a large number of layers. This explains the short time needed in recognizer engine. There are three free parameters: the number of convolution layers, hidden layers and neurons along feature axis. With Pure TDNN, the most number of neurons in the first and second convolution layers are set to 40 and 100 respectively (see Table 6). We notice that Pure TDNN is faster than SVM (12 mn Vs 20 mn) thanks to its constraints of weight sharing; receptive field and temporal delay which allow reducing the number of parameters (see Sect. 3). With hybrid TDNN-SVM, our system becomes slower than Pure TDNN and SVM but reached the most recognition rate. This result proves the effectiveness of this hybrid architecture and also demonstrates the relevance of the beta-elliptic approach.

Table 6. Most results of tests with Pure TDNN, SVM and hybrid TDNN-SVM

Architecture	Neurons in first conv. layer (feature axis)	Neurons in second conv. layer (feature axis)	Time (mn)	Recognition rate
Pure TDNN	40	100	12	95.16%
SVM (RBF kernel)	–	–	20	96.90%
Hybrid TDNN-SVM	40	150	27	99.52%

Table 7. Performance comparisons with other systems on LMCA database

Works	Architecture	Database	Recognition rate
[24]	MLP	LMCA	94.14%
[6]	Genetic algorithm		96.25%
[11]	SVM		97.51%
Present Work	**Hybrid TDNN-SVM**		**99.52%**

Compared to other works, hybrid TDNN-SVM is the more efficient in term of recognition rate, but we cannot compare its speed relative to the other systems because they do not mention the execution time (see Table 7).

6 Conclusion

We presented a new method for Arabic handwriting recognition based on hybrid TDNN-SVM architecture. In online handwriting field, the majority of researches use HMM and Recurrent Neural Network to present sequential input data. These systems require a big lot of computation unlike the TDNN which is faster due to its constraints of weight sharing and receptive field. The use of SVM after TDNN output layer allows enhancing the performance of our system. We have tested this architecture with beta-elliptic parameters and have showed an impressive recognition rate.

As perspective, we will add other geometric parameters to the beta-elliptic features and test our system on online Arabic word databases such as ADAB and IFN/ENIT.

References

1. Tagougui, N., Kherallah, M., Alimi, A.: Online Arabic handwriting recognition: a survey. Int. J. Doc. Anal. Recogn. (IJDAR) **16**(3), 209–226 (2013)
2. Chaabouni, A., Boubaker, H., Kherallah, M., Alimi, A.M.: Combining of off-line and on-line feature extraction approaches for writer identification. In: International Conference on Document Analysis and Recognition, pp. 1299–1303 (2011)
3. Maalej, R., Tagougui, N., Kherallah, M.: Online Arabic handwriting recognition with dropout applied in deep recurrent neural networks. In: 12th IAPR Workshop on Document Analysis Systems, pp. 417–421 (2016)
4. Abdelazeem, S., Hesham, M.: On-line Arabic handwritten personal names recognition based on HMM. In: International Conference on Document Analysis and Recognition System, pp. 1304–1308 (2013)
5. Nakkach, H., Haboubi, S., Amiri, H.: Online Arabic character recognition using global and local features. In: 3rd International Conference on Automation, Control, Engineering and Computer Science, pp. 120–124 (2016)
6. Kherallah, M., Bouri, F., Alimi, A.M.: On-line Arabic handwriting recognition system based on visual encoding and genetic algorithm. Eng. Appl. Artif. Intell. **22**, 153–170 (2009)
7. Charfi, M., Kherallah, M., El Baati A., Alimi, Adel M.: A new approach for Arabic handwritten postal addresses recognition. Int. J. Adv. Comput. Sci. Appl. **3**(3) (2012)

8. Daifallah, K., Jamous, H.: Recognition-based segmentation algorithm for on-line Arabic handwriting. In: 10th International Conference on Document Analysis and Recognition, pp. 886–890 (2011)
9. Tagougui, N., Boubaker, H., Kherallah, M., Alimi, A.M.: A hybrid NN/HMM modeling technique for online Arabic handwriting. Int. J. Comput. Linguist. Res. **4**(3), 107–118 (2013)
10. Hamdani, M., El Abed, H., Kherallah, M., Alimi, A.M.: Combining multiple HMMs using on-line and off-line features for off-line Arabic handwriting recognition. In: 10th International Conference on Document Analysis and Recognition, pp. 201–205 (2010)
11. Elleuch, M., Zouari, R., Kherallah, M.: Feature extractor based deep method to enhance online Arabic handwritten recognition system. In: Villa, A.E.P., Masulli, P., Pons Rivero, A. J. (eds.) ICANN 2016. LNCS, vol. 9887, pp. 136–144. Springer, Heidelberg (2016). doi:10.1007/978-3-319-44781-0_17
12. Boubaker, H., Kherallah, M., Alimi, A.M.: Handwriting and hand drawing velocity modeling by superposing beta impulses and continuous training component. Int. J. Comput. Sci. Issues **10**(5), 57–63 (2013)
13. Boubaker, H., Kherallah, M., Alimi, A.M.: New strategy for the on line handwriting modeling. In: 9th International Conference on Document Analysis and Recognition, vol. 2, pp. 1233–1247 (2009)
14. Kherallah, M., Haddad, L., Alimi, A.M.: On-line handwritten digit recognition based on trajectory and velocity modeling. Pattern Recogn. Lett. **29**(5), 580–594 (2008)
15. Plamondon, R., Alimi, A.M.: Modeling velocity profiles of rapid movements: a comparative study. Biol. Cybemetics **69**(2), 119–128 (1993)
16. Dhieb T., Ouarda W., Alimi, A.M.: Online Arabic writer identification based on beta-elliptic model. 15th International Conference on Intelligent Systems Design and Applications, pp. 74–79 (2015)
17. Waibel, A., Hinton, G.: Phoneme recognition using time delay neural network. IEEE Trans. Acoust. Speech Sig. Process. **37**(3), 328–339 (1989)
18. Ferrat, F., Guerti, M.: Classification of the Arabic emphatic consonants using time delay neural network. Int. J. Comput. Appl. **80**(10), 106–112 (2013)
19. Poisson, E., Lallican, P.: Multi-modular architecture based on convolutional neural networks for online handwritten character recognition. In: Proceedings of the 9th International Conference on Neural Information Processing, vol. 5, pp. 2444–2448 (2002)
20. Tagougui, N., Boubaker, H., Kherallah, M., Alimi, A.M.: Hybrid MLPNN/HMM recognition system for online Arabic handwritten script. In: World Congress on Computer and Information Technology, pp. 1–6 (2013)
21. Boubaker, H., Kherallah, M., Alimi, A.M.: Spatio-temporal representation of 3D hand trajectory based on beta-elliptic models. J. Inf. Assur. Secur. **18**(15), 1632–1647 (2016)
22. Elleuch, M., Maalej, R., Kherallah, M.: A new design based-SVM of the CNN classifier architecture with dropout for offline arabic handwritten recognition. Procedia Comput. Sci. **80**, 1712–1723 (2016)
23. Shigeo, A.: Fuzzy support vector machines for multi-label classification. Pattern Recogn. **48**(6), 2110–2117 (2015)
24. Boubaker, H., Elbaati, A., Tagougui, N., Kherallah, M., Alimi, A.M.: Online Arabic databases and applications. In: Märgner, V., El Abed, H. (eds.) Guide to OCR for Arabic Scripts, pp. 541–557. Springer, Heidelberg (2012)

3D Geometric Moment Invariants for ATS Drugs Identification: A More Precise Approximation

Satrya Fajri Pratama[1], Azah Kamilah Muda[1(✉)], Yun-Huoy Choo[1], and Ajith Abraham[1,2]

[1] Computational Intelligence and Technologies (CIT) Research Group, Center of Advanced Computing and Technologies, Faculty of Information and Communication Technology, Universiti Teknikal Malaysia Melaka, Hang Tuah Jaya, 76100 Durian Tunggal, Melaka, Malaysia
satrya@student.utem.edu.my, {azah,huoy}@utem.edu.my, ajith.abraham@ieee.org

[2] Machine Intelligence Research Labs (MIR Labs), Scientific Network for Innovation and Research Excellence, Auburn, WA, USA

Abstract. National development is constantly threatened by drug abuse. The chemical composition of the drugs heavily determines the results of identification process, which becomes more unreliable due to the introduction of new, sophisticated, and increasingly complex ATS analogues. The identification of the unique characteristics of molecular structure in ATS drug is very crucial. Therefore, this paper is meant for formulating a more precise 3D geometric moment invariants to represent the molecular structure. The performance of the proposed technique was analyzed using drug molecular structures obtained from United Nations Office of Drugs and Crime and also from various sources. The evaluation shows the technique is qualified to be further explored and adapted to be fully compatible with ATS drug identification domain.

Keywords: More precise geometric moments · 3D moments function · Moment invariants function · ATS drugs · Drugs identification · Molecular structure

1 Introduction

Every nation in the world possesses a common harrowing social problem, which is the abuse of Amphetamine-type Stimulants (ATS) drugs [1], including amphetamine, methamphetamine, and substances of the "ecstasy"-group (MDMA, MDA, MDEA, etc.). The struggles of finding concrete solution to prevent drugs abuse in society are encountered by national law enforcement authorities, due to the existence of new brand or unfamiliar ATS drug substances. Moreover, the development of chemical drugs with desirable biological effect is the main focus of cheminformatics research community, while less attention is given to the shape similarity search that can lead to identification of unknown substances.

A. Abraham et al. (eds.), *Proceedings of the 16th International Conference on Hybrid Intelligent Systems (HIS 2016)*, Advances in Intelligent Systems and Computing 552,
DOI 10.1007/978-3-319-52941-7_13

Commonly, the chemical composition of a drug or any substance in general allows it to be identified. However, the identification process becomes more unreliable due to the introduction of new, sophisticated, and increasingly complex ATS molecular structures. Furthermore, the limitations of the current test kit present a challenge to detect new brand or unfamiliar ATS drug to both national law enforcement authorities and scientific staff of forensic laboratories, while at the same time, it is prone to misidentification [2]. Due to these limitations, it is preferable to perform the identification by relying on the shape of molecular structures.

The shape of molecular structures basically can be represented by using both of 2-dimensional (2D) and 3-dimensional (3D) model, which is known as 2D and 3D shape descriptors. 2D shape descriptors have been developed which can be generally divided into boundary- and area-based. Meanwhile, 3D shape descriptors focus on volume- and surface-based descriptors. It also has been described as more powerful and accurately represent an object shape [3], in terms of providing more discriminative power compared to 2D shape descriptors. Thus, this paper believes that 3D descriptor can be used to identify molecular structure of ATS drug's chemical components, even for a new brand of ATS drug due to their similar ring substitutes.

This paper aims to propose a novel 3D Precise Geometric Moment Invariants to represent the ATS drug molecular structure. The remainder of the paper is organized as follows. An overview of ATS drug identification is given in the next section, while an overview of 3D Geometric Moment Invariants is provided in Sect. 3. In Sects. 4 and 5, the proposed technique is introduced and the experimental setup describing the data source collection and experimental design are presented respectively, while the results are discussed in Sect. 6. Finally, conclusion and future work is drawn in Sect. 7.

2 ATS Drug Identification

United Nations Office of Drugs and Crime (UNODC) have outlined a set of standard methods to conduct manual identification of ATS drugs. ATS are a group of substances, mostly synthetic in origin, that are structurally derived from β-phenethylamine [1]. Chemical modification at the positions R1 to R9, as shown in Fig. 1, results in a practically unlimited number of pharmacologically active compounds. Some of the notorious examples of ATS drugs are depicted in Fig. 2.

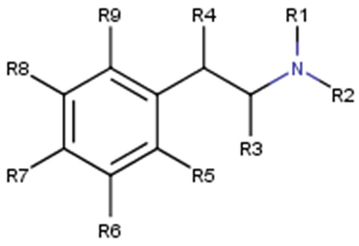


Fig. 1. β-phenethylamine

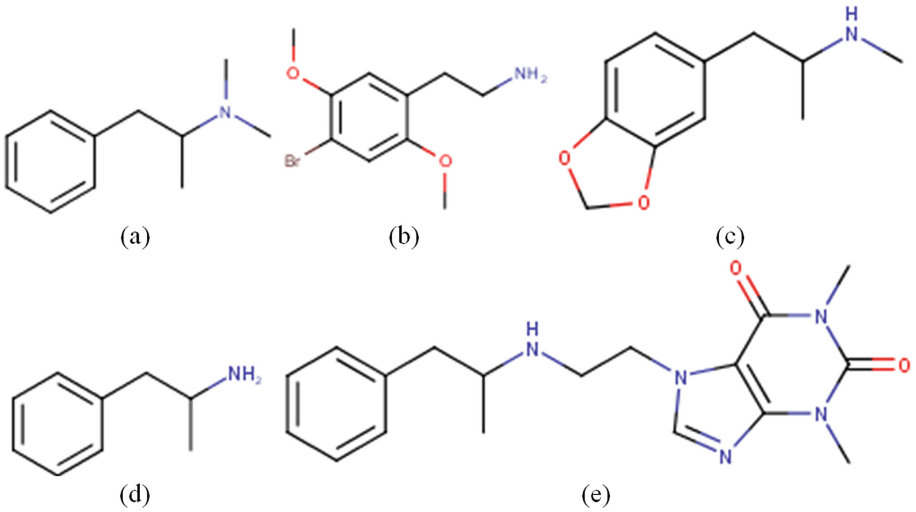


Fig. 2. Samples of ATS drug molecular structure: methamphetamine (a), 2C-B (b), MDMA (c), amphetamine (d), and fenetylline (e)

Different results are possibly obtained from different testing laboratory because of their non-conformance to these standards. Nevertheless, most of these laboratories agree to use gas chromatography/mass spectrometry (GC/MS) as the most common method to identify a chemical substance [1, 4, 5]. The flaws of GC/MS which are surfacing while trying to identify several ATS drugs are recently discovered in a study, most notably in identifying methamphetamine [6].

Methamphetamine itself has two stereo-isomers, which is *l*-methamphetamine and *d*-methamphetamine. Isomers are defined by [7] as one of several species (or molecular entities) that have the same atomic composition but different line or stereo-chemical formulae and hence different physical and/or chemical properties. GC/MS is also increasingly incapable to determine that several molecular structures are actually ATS drugs. While *l*-methamphetamine has very little pharmacodynamics effect, *d*-methamphetamine on the other hand is a controlled substance that has high potential for abuse and addiction [8].

As mentioned earlier, this paper believes that by relying on the shape of the molecular structure of the drug itself, the identification process can be improved. Both 2D and 3D molecule structures are basically represented as shape image. The molecular representation is modeled from molecules and enables mathematical treatment, and these representations are commonly referred as molecular descriptors [9]. There are simple molecular descriptors, usually called topological or 2D-descriptors, and there are molecular descriptors derived from a geometrical representation that are called geometrical or 3D-descriptors.

Geometrical descriptors usually provide more information and discrimination power than topological descriptors because it involves knowledge of the relative positions of the atoms in 3D space for similar molecular structures and molecule

conformations [10]. Moreover, topological descriptor hides the properties of the ring substitutes in a molecule. Therefore, this paper proposes a new geometrical descriptor from volume data of 3D molecular structure, which is essential to show and differentiate the unique features at a ring substitute.

3 Existing 3D Geometric Moment Invariants

Moments Function (MF) can be used to generate a set of moments that uniquely represent the global characteristic of an image. Moments are scalar quantities used to characterize a function and to capture its significant features. Moment Invariants are very useful tools for pattern recognition [11].

Moment Invariants (MI) was first introduced to pattern recognition and image processing through the employment of algebraic invariants theory by [12]. And thus ever since, it has been chosen as one of the most important and frequently used shape descriptors options, and has been extended to 3D images as well [13]. Even though they suffer from certain intrinsic limitations, they frequently serve as first-choice descriptors and a reference method for evaluating the performance of other shape descriptors [13]. 3D Geometric Moments (GM) of image intensity function $f(x, y, z)$ is generally defined as [13]

$$m_{pqr} = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} x^p y^q z^r f(x, y, z) dxdydz \tag{1}$$

where $p, q, r = 0, 1, 2, \ldots$ The sum of the indices is called the order of the moment. The translation invariance can easily be provided by the 3D Central Geometric Moments [14]

$$\mu_{pqr} = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} (x - x_c)^p (y - y_c)^q (z - z_c)^r f(x, y, z) dxdydz \tag{2}$$

where the center of gravity (centroid) of the image intensity function $f(x, y, z)$ is calculated by $x_c = \frac{m_{100}}{m_{000}}$, $y_c = \frac{m_{010}}{m_{000}}$, and $z_c = \frac{m_{001}}{m_{000}}$. The central moments are often normalized to produce scale invariants [14]

$$\eta_{pqr} = \frac{\mu_{pqr}}{\mu_{000}^{(p+q+r+3)/3}} \tag{3}$$

Recently, [15] proposed novel 3D rotation invariants from GM using moment tensor method, and also implemented an automatic method for generating those 3D rotation invariants. These invariants are built up from the moments of order 2 up to order 16, and consist of 1185 invariants. The first 2 rotation invariants are presented below

$$I_1 = (\mu_{200} + \mu_{020} + \mu_{002}) / \mu_{000}^2 \tag{4}$$

$$I_2 = \left(\mu_{200}^2 + \mu_{020}^2 + \mu_{002}^2 + 2\mu_{110}^2 + 2\mu_{101}^2 + 2\mu_{011}^2\right) / \mu_{000}^4 \tag{5}$$

The zeroth-order moment (m_{000}) is used as divisor to normalize the GM with respect to scaling. This technique is hereby dubbed as 3D Suk–Flusser Rotation Invariants Central Geometric Moments (3D SFRI-CGM).

4 Proposed 3D Precise Geometric Moment Invariants for Molecular Structure Representation

A digital 3D image of size $N \times N \times N$ is an array of voxels (volume pixels), therefore the triple integral in (1) must be replaced by triple summation. The most common way is to employ the rectangular, i.e., zero-order method of numeric integration. And thus, (1) takes the following discrete form:

$$\hat{m}_{pqr} = \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{k=1}^{N} i^p j^q k^r f_{ijk} \tag{6}$$

where i, j, k are coordinates of the voxels and f_{ijk} is the gray-level of the voxel i, j, k. It should be noted that $\hat{m}_{pqr}$ is just an approximation of m_{pqr}. However, [16] and later generalized by [17], proposes a formula to obtain a more precise estimation for calculating 2D GM, where the authors integrate the monomials $x^p y^q$ precisely by the Newton–Leibnitz formula on each pixel

$$\dot{m}_{pq} = \sum_{i=1}^{N}\sum_{j=1}^{N} f_{ij} \iint_{A_{ij}} x^p y^q dxdy = \sum_{i=1}^{N}\sum_{j=1}^{N} U_p(i) U_q(j) f_{ij} \tag{7}$$

$$U_s(a) = \frac{(a+0.5)^{s+1} - (a-0.5)^{s+1}}{s+1} \tag{8}$$

This study proposes the extension of 2D PGM into 3D PGM and thus (7) takes the following form:

$$\dot{m}_{pqr} = \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{k=1}^{N} f_{ijk} \iiint_{A_{ijk}} x^p y^q z^r dxdydz = \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{k=1}^{N} U_p(i) U_q(j) U_r(k) f_{ijk} \tag{9}$$

where A_{ij} and A_{ijk} denotes the area of the pixel (i, j) and voxel (i, j, k) respectively. Equations (7) and (9) are still a zeroth-order approximation of moments of the original image is approximated, the monomials are integrated exactly. Thus a more precise GM is obtained, hence it is known as Precise Geometric Moments (PGM). The idea of this paper is to incorporate PGM instead of regular GM prior to the formulation of 3D Suk–Flusser Rotation Invariants.

It is also necessary to produce translation invariants PGM by calculating its central moments. A relationship between central and non-central moments for regular GM was provided by [18], which defines a convenient formula to calculate central moments from non-central moments, and vice versa

$$\mu_{pqr} = \sum_{a=0}^{p}\sum_{b=0}^{q}\sum_{c=0}^{r}\binom{p}{a}\binom{q}{b}\binom{r}{c} m_{100}^{p-a} m_{010}^{q-b} m_{001}^{r-c} (-m_{000})^{-(p+q+r-a-b-c)} m_{abc} \quad (10)$$

$$m_{pqr} = \sum_{a=0}^{p}\sum_{b=0}^{q}\sum_{c=0}^{r}\binom{p}{a}\binom{q}{b}\binom{r}{c} m_{100}^{p-a} m_{010}^{q-b} m_{001}^{r-c} m_{000}^{-(p+q+r-a-b-c)} \mu_{abc} \quad (11)$$

Even though (10) and (11) were meant for regular GM, this paper proposes to employ (10) as a mean to calculate Central PGM. And thus, it is an efficient and elegant way of deriving rotation invariants from PGM in 3D based on 3D CPGM, namely 3D Suk–Flusser Rotation Invariants Central Precise Geometric Moments (3D SFRI-CPGM). This is the main theoretical contribution of the paper. In the next section, the invariance property on ATS and non-ATS dataset is demonstrated, and shows the numerical stability of both 3D SFRI-CGM and 3D SFRI-CPGM.

5 Experimental Setup

With the goal stated in the section above, an empirical comparative study must be designed and conducted extensively and rigorously. A detailed description of the experimental method is provided in this section.

5.1 Data Source Collection

This section describes the process of transforming molecular structure of ATS drug into 2D and 3D computational data representation. ATS dataset used in this research comes from [1], which contains 60 molecular structures which are commonly distributed for illegal use. On the other hand, 60 non-ATS (n-ATS) drug molecular structures are also collected from various sources will be used as benchmarking dataset.

These structures are drawn in 2D molecular structure format using MarvinSketch 15.11.9.0 [19]. After the 2D molecular structure is created, the structure will be cleaned and transformed to 3D molecular structure, also by using MarvinSketch. The structure will be then saved as MDL MOL file. The MDL MOL file must be then converted to Virtual Reality Markup Language (VRML) format, because VRML format is the input type required for generating voxel data of 3D molecular structure. In order to convert SDF file to VRML file, Jmol 14.4.0 [20] is required.

VRML file will be then voxelized to voxel grid data with 512 voxel resolution using binvox 1.21 program [21] for the training dataset, and will be uniquely and randomly translated and rotated 50 different times for the testing dataset. After the voxel data has been generated, 3D SFRI-CGM and 3D SFRI-CPGM for training and testing dataset will be calculated from 16th order using existing and proposed technique respectively. The sample of the voxelized molecular structure is shown in Fig. 3 and the output of the existing and proposed technique is shown in Table 1, respectively, which consists of 1185 invariants values. The molecular structure dataset for all formats (MDL MOL, VRML, and BINVOX) and computed 3D SFRI-CGM and 3D SFRI-CPGM are publicly available in [22].

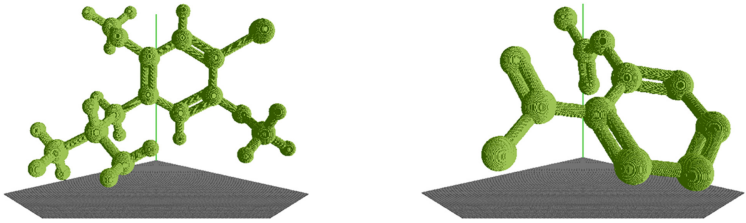


Fig. 3. Voxelized molecular structure of an ATS drug, 2C-I (a) and n-ATS drug, Aspirin (b).

Table 1. Sample output of 3D MI techniques

Structure name	3D MI	I1	I2	…	I1184	I1185
2C-I	SFRI-CGM	22.21497	15.49400	…	65536.87	6.361E09
	SFRI-CPGM	22.21510	15.49406	…	65539.87	6.362E09
Aspirin	SFRI-CGM	21.11886	14.64512	…	72363.25	9.107E09
	SFRI-CPGM	21.11898	14.64518	…	72366.04	9.108E09

5.2 Experimental Design

The traditional framework of pattern recognition tasks, which are preprocessing, feature extraction, and classification, will be employed in this paper. This paper will compare the performance of 3D SFRI-CGM and 3D SFRI-CPGM. It is also worth mentioning that all MIs are magnitude normalized to their respective degree.

All extracted instances are tested using training and testing dataset aforementioned for its invariance on the same ATS molecular structure, and also classification of unknown molecular structure, all of which are executed for 50 times (50-fold cross validation). In order to justify the quality of features produced by each MF, the features are tested against well-known classifier, Random Forest (RF) [23] from WEKA Machine Learning package [24]. RF is employed in this study because previous studies conducted by [25, 26] have found that RF is the most suitable for the molecular structure data. In this study, the number of trees employed by RF is 1185, equals to the number of attributes of both techniques.

However, the processing time and the memory consumption of the existing and proposed technique are not the focus of this study and thus the results are omitted. It should also be mentioned that this paper focuses on the improvement on the computation of moment invariants, therefore the comparison against existing 3D molecular descriptors are currently absent in this study.

6 Experimental Results and Discussions

In this section, the proposed invariants will be tested numerically by constructing 3D SFRI-CGM and 3D SFRI-CPGM from invariant 1 to 1185 for all 60 ATS drugs molecular structure. The experiments are designed to verify rotation invariance and evaluate the numerical stability of the proposed invariants.

6.1 Invariance on the Same ATS Molecular Structure Analysis

To evaluate quantitatively the invariance, mean relative error (MRE) is used to measure the computational error of the *i*th invariant. The MRE of the *i*th invariant is defined as

$$MRE_i = \frac{1}{N}\sum_{j=i}^{N}\left|\frac{I_i^j - I_i}{I_i}\right| \times 100\% \tag{12}$$

where I_i and I_{ij} are the *i*th invariants of the original molecular structure and the *j*th rotated version, respectively, and N is the number of rotated versions. Consequently, it is found the average MRE (AMRE) and maximum MRE (MMRE) for each molecular structure. AMREs and MMREs from both 3D SFRI-CGM and 3D SFRI-CPGM exhibit the same result. AMREs for all 60 ATS molecular structures are below 1%, while the value of MMREs is shown in Fig. 4. But nevertheless, this experiment exhaustively demonstrated desirable rotation invariance.

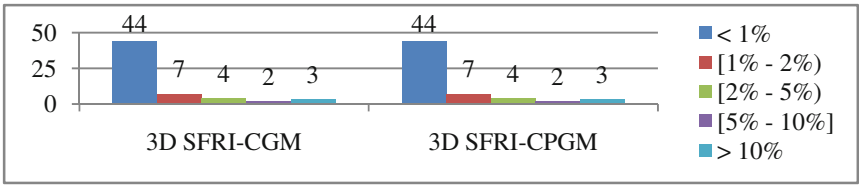


Fig. 4. Maximum Mean Relative Errors of 60 molecular structures for 3D SFRI-CGM and 3D SFRI-CPGM

6.2 Classification of Unknown ATS Drug Molecular Structure

The comparison of classification accuracy of 3D SFRI-CGM and 3D SFRI-CPGM is also one of the primary considerations of this paper. Table 2 shows the results of mean classification accuracy from 50 executions using RF classifier.

Table 2. Mean classification accuracy for 3D MI techniques

3D MI	Mean accuracy
SFRI-CGM	88.87%
SFRI-CPGM	90.00%

Based on the results shown in Table 2, it is evident that 3D SFRI-CPGM produces slightly better result of mean classification accuracy. However, to further validate the strength of 3D SFRI-CPGM compared to 3D SFRI-CGM, in-depth statistical validation using independent samples *t*-test must be conducted, by using SPSS 17 software. Independent samples *t*-test was selected because the result of the test of normality on the classification accuracy of both techniques using Shapiro-Wilk test showed that they are normally distributed, and thus parametric testing are suitable to be conducted. There

was a statistically significant difference in the accuracy using 3D SFRI-CGM ($\mu = 0.8887$, $\sigma = 0.00785$) and 3D SFRI-CPGM ($\mu = 0.90$, $\sigma = 0.0$); $t(49) = -10.204$, $p = 0.0$. This result suggests that the 3D SFRI-CPGM is capable to differentiate the unique features of ATS drugs and n-ATS drugs at a ring substitute.

7 Conclusions and Future Works

This paper proposed an incorporation of 3D CPGM into 3D Suk–Flusser Rotation Invariants and compared the merits of the proposed technique with 3D SFRI-CGM. The experiments have shown that the proposed technique produces statistically significant better results compared to the existing one. And thus, an extensive comparative study on existing and proposed MI techniques for representing drug molecular structure has been presented.

Hence, future works are to incorporate specifically-tailored classifiers for shape representation, and ATS drug molecular structure data from National Poison Centre, Malaysia, will also be used as additional dataset in the future works. Extensive comparative study of the proposed technique against existing 3D molecular descriptors will also be conducted in the future works.

Acknowledgements. This work was supported by UTeM Postgraduate Fellowship Scheme and PJP High Impact Research Grant (S01473-PJP/2016/FTMK/HI3) from Universiti Teknikal Malaysia Melaka (UTeM), Malaysia.

References

1. United Nations Office of Drugs and Crime: Recommended Methods for the Identification and Analysis of Amphetamine, Methamphetamine and Their Ring-Substituted Analogues in Seized Materials, vol. Sales No. E.06.XI.1. UNODC, New York (2006)
2. Saitman, A., Park, H.-D., Fitzgerald, R.L.: False-positive interferences of common urine drug screen immunoassays: a review. J. Anal. Toxicol. **38**(7), 387–396 (2014). doi:10.1093/jat/bku075
3. Flusser, J., Suk, T., Zitová, B.: 2D and 3D Image Analysis by Moments, vol. 1. Wiley, West Sussex (2016)
4. Langman, L.J., Bowers, L.D., Collins, J.A., Hammett-Stabler, C.A., LeBeau, M.A.: Gas Chromatography/Mass Spectrometry Confirmation of Drugs; Approved Guidelines, 2nd edn. Clinical and Laboratory Standards Institute, Pennsylvania (2010)
5. Lin, D.-L., Yin, R.-M., Ray, L.H.: Gas Chromatography-Mass Spectrometry (GC-MS) analysis of amphetamine, methamphetamine, 3,4-methylenedioxyamphetamine and 3,4-methylenedioxymethamphetamine in human hair and hair sections. J. Food Drug Anal. **13**(3), 193–200 (2005)
6. McShane, J.J.: GC-MS is Not Perfect: The Case Study of Methamphetamine (2011)
7. International Union of Pure and Applied Chemistry: Compendium of Chemical Terminology. Gold Book, 2nd edn. Blackwell Scientific Publications, Oxford (2006)

8. Mendelson, J., Uemura, N., Harris, D., Nath, R.P., Fernandez, E., Jacob, P., Everhart, E.T., Jones, R.T.: Human pharmacology of the methamphetamine stereoisomers. Clin. Pharmacol. Ther. **80**(4), 403–420 (2006)
9. Todeschini, R., Consonni, V.: Descriptors from molecular geometry. In: Handbook of Chemoinformatics, pp. 1004–1033. Wiley-VCH Verlag GmbH (2008)
10. Consonni, V., Todeschini, R.: Molecular Descriptors. In: Puzyn, T., Leszczynski, J., Cronin, T.M. (eds.) Recent Advances in QSAR Studies: Methods and Applications, pp. 29–102. Springer, Dordrecht (2010)
11. Sun, Y., Liu, W., Wang, Y.: United moment invariants for shape discrimination. In: International Conference on Robotics, Intelligent Systems and Signal Processing, Changsha 2003, pp. 88–93. IEEE (2003)
12. Hu, M.-K.: Visual pattern recognition by moment invariants. IRE Trans. Inf. Theory, **8**(2) 179–187 (1962). doi:10.1109/TIT.1962.1057692
13. Sadjadi, F.A., Hall, E.L.: Three-dimensional moment invariants. IEEE Trans. Pattern Anal. Mach. Intell. **PAMI-2**(2), 127–136 (1980). doi:10.1109/TPAMI.1980.4766990
14. Flusser, J., Suk, T., Zitová, B.: Moments and Moment Invariants in Pattern Recognition, vol. 1. Wiley, West Sussex (2009)
15. Suk, T., Flusser, J.: Tensor method for constructing 3D moment invariants. In: Real, P., Diaz-Pernil, D., Molina-Abril, H., Berciano, A., Kropatsch, W. (eds.) CAIP 2011. LNCS, vol. 6855, pp. 212–219. Springer, Heidelberg (2011). doi:10.1007/978-3-642-23678-5_24
16. Lin, W.-G., Wang, S.-S.: A note on the calculation of moments. Pattern Recogn. Lett. **15** (11): 1065–1070 (1994). http://dx.doi.org/10.1016/0167-8655(94)90121-X
17. Flusser, J.: Refined moment calculation using image block representation. IEEE Trans. Image Process. **9**(11), 1977–1978 (2000). doi:10.1109/83.877219
18. Xu, D., Li, H.: Geometric moment invariants. Pattern Recogn. **41**(1), 240–249 (2008)
19. ChemAxon Ltd.: Marvin (2015). http://www.chemaxon.com
20. Jmol: an open-source Java viewer for chemical structures in 3D (2015). http://www.jmol.org/
21. Min, P.: Binvox 3D mesh voxelizer, Princeton University (2015). http://www.cs.princeton.edu/~min/binvox
22. Computational Intelligence and Technologies Research Lab: Amphetamine-type stimulants drug molecular structure dataset (2016). http://ftmk.utem.edu.my/cit/downloads/ats-drugs
23. Breiman, L.: Random Forests. Mach. Learn. **45**, 5–32 (2001)
24. Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H.: The WEKA data mining software: an update. SIGKDD Explor. **11**, 10–18 (2009)
25. Pratama, S.F., Muda, A.K., Choo, Y.-H., Abraham, A.: A comparative study of 2D UMI and 3D Zernike shape descriptor for ATS drugs identification. In: Abraham, A., Muda, A.K., Choo, Y.-H. (eds.). AISC, vol. 355, pp. 237–249Springer, Heidelberg (2015). doi:10.1007/978-3-319-17398-6_22
26. Pratama, S.F., Muda, A.K., Choo, Y.-H., Abraham, A.: Exact computation of 3D geometric moment invariants for ATS drugs identification. In: Snášel, V., Abraham, A., Krömer, P., Pant, M., Muda, A.K. (eds.). AISC, vol. 424, pp. 347–358Springer, Heidelberg (2016). doi:10.1007/978-3-319-28031-8_30

An Efficient Dynamic Priority-Queue Algorithm Based on AHP and PSO for Task Scheduling in Cloud Computing

Hicham Ben Alla[1(✉)], Said Ben Alla[1], Abdellah Ezzati[1], and Abdellah Touhafi[2]

[1] LAVETE Laboratory, Science and Technical Faculty, Mathematics and Computer Science Department, Hassan 1 University, Settat 26000, Morocco
hich.benalla@gmail.com, saidb_05@hotmail.com, abdezzati@gmail.com

[2] Department of Electronics and Informatics (ETRO), Vrije Universiteit Brussel, Pleinlaan 2, 1050 Brussels, Belgium
abdellah.touhafi@vub.ac.be

Abstract. Nowadays Cloud Computing is an emerging technology in the area of parallel and distributed computing. Task scheduling is one of the major issues in cloud computing, which plays an important role to improve the overall performance and services of the cloud. Task scheduling in cloud computing means assign best suitable resources for the task to be executed with the consideration of different parameters like execution time, user priority, cost, scalability, throughput, makespan, resource utilization and so on. In this paper, we address the challenge of task scheduling, and we consider one of most critical issues in scheduling process such as the task priorities. The goal of this paper is to propose an efficient Dynamic Priority-Queue (DPQ) algorithm based on Analytic Hierarchy Process (AHP) with Particle Swarm Optimization (PSO) algorithm. The proposed algorithm DPQ-PSO gives full consideration to the dynamic characteristics of the cloud computing environment. Further, the proposed algorithm has been validated through the CloudSim simulator. The experimental results validate that the proposed approach can effectively achieve good performance, user priority, load balancing, and improve the resource utilization.

Keywords: Cloud computing · PSO algorithm · AHP · DPQ-PSO algorithm · Task scheduling · Dynamic queues

1 Introduction

The concept of Cloud Computing has appeared with the rapid development of processing and storage technologies and the success of the Internet. According to the NIST [1], the Cloud Computing refers to the concept of allowing users to request a variety of services like storage, computing power, applications at anytime, anywhere and in any quantity, on the basis that pay only what they use. Scheduling is an interesting challenge in cloud computing nowadays. According to the needs of optimal allocation of resources

A. Abraham et al. (eds.), *Proceedings of the 16th International Conference on Hybrid Intelligent Systems (HIS 2016)*, Advances in Intelligent Systems and Computing 552,
DOI 10.1007/978-3-319-52941-7_14

and achieving QoS. These resources should be utilized properly and efficiently, and assign the appropriate resource to the task. Scheduling problem for user requests in cloud computing environment is NP-complete problem. This problem is usually solved by using heuristic methods in order to reduce to polynomial complexity [2]. In this process, the cloud scheduler receives the tasks from the users and maps them to available resources taking into consideration tasks' attributes, and requirements such as length, deadline, waiting time etc., and the resource parameters and properties. So, to address this challenge, there is a need to design priority-based task scheduling approach in cloud environment that aim to achieve an advantage performance so as to reduce the makespan and execution time and gives full consideration to the characteristics of tasks. Hence, this paper proposes an efficient Dynamic Priority-Queue algorithm (DPQ) based on Analytic Hierarchy Process (AHP) and particle swarm optimization (PSO) algorithm. In general, the proposed work aims to prioritize the task list based on multiple criteria into dynamic queue and assign an appropriate resource to the task. The remainder of the paper is organized as follows: Sect. 2 presents related works. In Sect. 3, the proposed work is described. Section 4 discusses the experiment setup and simulation results of the proposed work. Finally, the paper gives a conclusion in Sect. 5.

2 Related Work

This section contains overview of some existing algorithms and approaches related to the proposal presented in this paper. In the paper [3], an improved scheduling algorithm is introduced based on user priority and task length (size). Based on these two criteria order of execution of the tasks will be decided by the datacenter broker. In the paper [4], the key role of the algorithm proposed is the QOS-driven based on the priority of task which in turn is computed using many task attributes such as user privilege, expectation, and the length of task, next, this task scheduled onto the service which has a minimum completion time. In the paper [5], Authors propose multi queue scheduling (MQS) algorithm to reduce the cost of both reservation and on-demand plans using the global scheduler. The proposed methodology depicts the concept of clustering the jobs based on burst time. The optimality is attained in this method by choosing jobs dynamically in order to achieve the optimum cloud scheduling problem and efficiently utilize the unused free space economically. Authors in the paper [6], propose an algorithm to schedule workflow tasks over the available cloud resources that minimize the execution cost and the execution time while meeting the deadline and budget constraints. Each workflow's task is assigned priority using bottom level. These priorities are then used to initialize the PSO. In the paper [7], the authors proposed an improvement in priority based job scheduling algorithm in cloud computing which is based on multiple criteria and multiple attribute decision making model using Analytical Hierarchy Process (AHP). This algorithm uses an iterative method to find priority of jobs and resources to achieve better performance. The authors in the paper [8], presents an algorithm which allow to each job requests a resource with determined priority. The proposed algorithm is based on AHP model as a multiple criteria decision making model, and consisted of three levels of priorities including scheduling level, resources level and job level.

3 Proposed Work

3.1 Task Scheduling and Problem Description

In cloud computing environments, we always face a wide variety of attributes that should be considered [9]. It means a particular task scheduling algorithm in cloud environments should pay attention to the properties of tasks. However, there are some criteria to be taken into consideration, so these parameters can be considered as single objective or multi-objective simultaneously. Priority of tasks is an important issue and need to be solved in scheduling process because some tasks need to be serviced earlier than other remaining tasks that can stay for a long time. Therefore, a suitable task scheduling algorithm must consider the priority of tasks based on multi-criteria. The makespan is another important criteria, it refers to the time spent for executing all tasks. In fact, the makespan has a direct effect on utilization of resources. In other words, when the utilization increases, the makespan surely decrease. Thus, an optimal tasks scheduling algorithm based on these parameters should be implemented in the Cloud broker. On the basis of issues mentioned above, the main objectives of the proposed work are:

i. Compute the priority of tasks based on multiple criteria decision making model using Analytical Hierarchy Process (AHP). Then, the quartile method has been adopted to classify tasks and resources in different level priority.
ii. Dispatch the tasks among dynamic Priority-Queues based on decision distribution and priority level.
iii. Scheduling the tasks stored in the dynamic queues using a meta-heuristic algorithm such Particle Swarm Optimization (PSO) in order to achieve good performance.

3.2 The Analytic Hierarchy Process (AHP)

The Analytic Hierarchy Process (AHP) is the well-know and popular method of Multi-criteria Decision Making (MCDM). MCDM refers to making decisions in the presence of multiple, usually conflicting criteria [10]. AHP is a simple, flexible and practical multi-criteria decision making model, developed by Saaty [11]. AHP firstly decompose the decision problem into a hierarchy of more easily comprehended sub-problems, each of which can be analyzed independently. Basically architecture of AHP is consisted of three levels which are objective level, attributes level and alternatives level respectively. The foundation of AHP is comparison matrix which can be shown as Eq. (1).

$$A = \begin{cases} a_{ij} = \frac{1}{a_{ij}} & i \neq j \\ 1 & i = j \end{cases} \tag{1}$$

Each entry in the matrix A is positive (a_{ij}). Also A is a square matrix $A_{n \times n}$ [8]. For any arbitrary comparison matrix such as A we can compute a vector of weights

$\omega = (\omega_{1,}\omega_{2,}\ldots,\omega_n)$ such as associated with A. Relationship between A and ω can be shown as Eq. (2).

$$A = \begin{cases} a_{ij} = \frac{\omega_i}{\omega_j} & i \neq j \\ 1 & i = j \end{cases} \tag{2}$$

The Vector of weights can be computed through the Eq. (3).

$$A\omega = \lambda_{\max}\omega \tag{3}$$

In Eq. (3) $\lambda_{\max}$ max is the principal eigenvalue of A and ω [8, 11, 12]. If A is absolutely consistent then the value of $\lambda_{\max} = n$. Saaty has defined Consistency Ratio (CR) as shown in Eq. (4).

$$CR = \frac{CI}{RI} \tag{4}$$

In Eq. (4) *RI* is the random index and it can be calculated randomly based on rank of comparison matrix [8, 13]. *RI* is calculated by Saaty [11] and indicated that the comparison matrix will be consistent if CR < 0.1. Also Consistency Index *CI* in Eq. (4) can be calculated by Eq. (5).

$$CI = (\lambda_{\max} - n)/(n-1) \tag{5}$$

3.3 Particle Swarm Optimization (PSO)

Particle swarm optimization (PSO) is a population-based stochastic optimization technique was first introduced in 1995 [14]. Each particle represents a candidate solution to the optimization problem. The advantages of PSO over many other optimization algorithms are its simplicity in implementation and its ability to converge to a reasonably good solution quickly. We consider that the search space is d-dimensional. In every iteration, each particle is updated by following two position values, P_i called personal best (P_{best}), is the best position achieved so long by particle i and P_g called global best (G_{best}), is the best position found by the neighbors of the particle i. After finding the two best values, the particle updates its velocity and positions with following Eqs. (6) and (7):

$$v_i^{t+1} = \omega.v_{id}^t + c_1.r_1.\left(p_i^t - x_i^t\right) + c_2.r_2.\left(p_g^t - x_i^t\right) \tag{6}$$

$$x_i^{t+1} = x_i^t + v_i^{t+1} \tag{7}$$

Where v_i^t and x_i^t are the component in dimension d of the i^{th} particle velocity and position in iteration t respectively. c_1, c_2 are constant weight factors, r_1, r_2 are random factors in the [0,1] interval and ω is the Inertia weight. The parameters ω, c_1 and c_2 must

be selected properly for increasing the capabilities of PSO algorithm [15]. Many strategies have been proposed to choose the proper value of ω, such as Chaotic Inertia Weight [16], The Linearly Decreasing Inertia Weight strategy (LDIW) [17], and Random Inertia Weight (RIW) [18]. In this paper, we use the RIW method as it can achieve best convergence velocity and precision, and can help to keep swarm variety. However, the original PSO version is designed for continuous function optimization problems, not for discrete function optimization problems. So, a binary version of PSO (BPSO) algorithm was developed to solve discrete function optimization problems [19]. The logistic sigmoid function shown in (8) can be used for the needs that the probability stays in the range of [0,1], in another word, it used to limit the speed of the particle:

$$S\left(v_i^{t+1}\right) = \frac{1}{1 + e^{-v_i^{t+1}}} \tag{8}$$

The equation that updates the particle position becomes the following:

$$x_i^{t+1} = \begin{cases} 1 & if \quad r_3 < S\left(v_i^{t+1}\right) \\ 0 & otherwise \end{cases} \tag{9}$$

Where r_3 is a random factor in the [0,1] interval.

3.4 Proposed Scheduling Approach

The motivation of the proposed work is to improve scheduling by assigning tasks based on dynamic priorities to the best suitable resource selected using meta-heuristic algorithm. In the entire of the system as shown in Fig. 1, the tasks are stored in the global queue according to their arrival time. Then, AHP model is used in order to prioritize tasks among the global priority-queue. Next, we apply a Dynamic Priority-Queues algorithm (DPQ). This algorithm in general aims to manage the global priority queue automatically by classifying tasks into three level priorities (Low, Medium, and High). Then dispatch the tasks among dynamic queue. The task having higher priority is considered as a most urgent task to be scheduled. The classification is obtained based on quartile method. This technique aims to divide all tasks into three quartiles. The lower quartile is the value of the middle of the first set, where 25% of the values are smaller than the first quartile and 75% are larger. The median quartile is the middle value of the data set. The upper quartile is the value of the middle of the second set, where 75% of the values are smaller than the third quartile and 25% are larger.

When DPQ algorithm dispatch all tasks among dynamic priority-queue, then the scheduler receives all queues generated, DPQ selects each queue and schedule tasks to appropriate resource (Vms) on the basis of PSO algorithm. The objective is executing tasks on the available resources with the minimum makespan. So the fitness function in (10) is used to calculate the executions times of all possible tasks sequences on every cloud resource, then, return the maximum value. The scheduling of tasks using the PSO algorithm can effectively maximize the resource utilization and reduce the makespan as well as the cost of using resources.

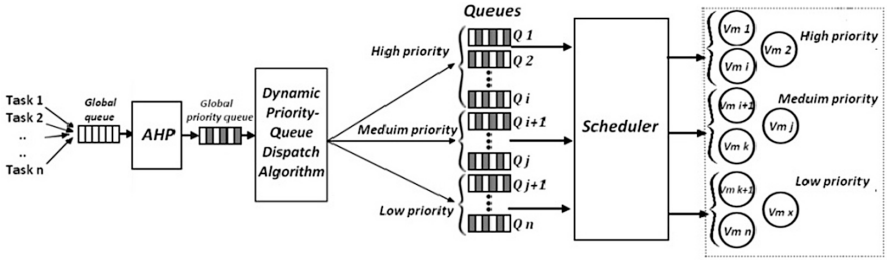


Fig. 1. Scheduling process of DPQ-PSO

$$\text{Fitness} = \text{Max}\left\{\varphi_{vm_1}, \varphi_{vm_2}, \ldots, \varphi_{vm_j}, \ldots\ldots\ldots, \varphi_{vm_m}\right\} \tag{10}$$

Where $\varphi_{vm_j} = \sum_{i=0}^{n} \delta_j(Task_i), \delta_j(Task_i)$ is the execution time of task i on vm_j, n is the number of tasks and φ_{vm_j} is the total execution time of set of tasks running on vm_j.

In Fig. 2, the AHP model consists of three levels. The objective level is to get best priority under many different parameters. The attribute level is composed of task parameters or decision criteria. The most important criteria are considered such as task length, waiting time, burst time and deadline. The decision alternatives represented by tasks in the alternative level. A comparison matrix can be built based on the Saaty Rating Scale [20], which is used to determine the relative importance of each task in terms of each criterion. The weights of all tasks can be derived using the analysis hierarchy process (AHP). Consider that $T = \{T1, T2, T3, \ldots, Ti\}$ be a set of tasks in cloud environment and $P = \{P1, P2, P3, \ldots, Pj\}$ be a set of task parameters. Assume that $C^1, C^2, C^3, \ldots, C^d$ are d comparison matrixes of tasks which are created according to priority of task parameters. For each of comparison matrixes we should compute a priority vector (vector of weights). The priority vector can be obtained by solving Eq. (3). There are several methods for calculating priority vector [8, 11, 20]. An iterative method for solving Eq. (3) can be found in [21]. This method solves the Eq. (3) by using numerical methods. In this case we can define a normal matrix of tasks level $\Delta = [\omega^1, \omega^2, \omega 3, \ldots, \omega^d]$. It is clear that Δ is a matrix with m (the number of tasks) rows and d (the number of parameters) columns. The next step is to make a comparison matrix for resources according to priorities. This matrix determines that which resource has higher priority than others based on decision maker(s). In this case, we will have a matrix with d rows and d columns. Assume that M is comparison matrix for task attributes, thus γ will be defined as priority vector of M. The next step of the algorithm is to calculate PVM which is denoted as priority vector of prioritizing tasks. PVM can be calculated by Eq. (11).

$$PVM = \Delta.\gamma \tag{11}$$

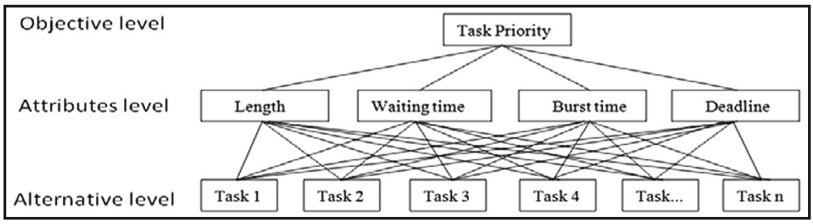


Fig. 2. AHP model of task priority

4 Experimental Results

To evaluate the proposed algorithm, the simulation is implemented using CloudSim simulator [22]. This simulator supports both modeling and simulation for single and inter-networked clouds. CloudSim enables seamless modeling, experiments and simulation of the cloud computing systems and application provisioning environments. The simulation is done using 10 Datacenter, [2–6] hosts, 30 virtual machines where processor MIPS in [1000–30000], Ram memory in [256–2048] and the Bandwidth in [500–1000]. For the tasks source, we get tasks from workload data. We use workload traces from real systems available from the Parallel Workload Archive (PWA). The workload data used is called The Cornell Theory Center (CTC) IBM SP2 log [23].

4.1 Numeric Example

The following is an example of the AHP process to prioritize the tasks in the proposed scheduling algorithm. We assume that three tasks are received to be handled by the Cloud Provider. All the tasks are of different nature depending on the requirement of Length, waiting time, burst time and deadline. The Relative priority matrix of task attributes is shown in Table 1.

Comparison matrix can be shown as follows:

$$M = \begin{pmatrix} 1 & \frac{1}{3} & \frac{1}{4} & \frac{1}{2} \\ 3 & 1 & 3 & 4 \\ 4 & \frac{1}{3} & 1 & 3 \\ 2 & \frac{1}{4} & \frac{1}{3} & 1 \end{pmatrix}$$

M is consistent because we have: $\lambda_{\max} = 4.30$, $CI = \frac{4.30-4}{4} = 0.076$, $CR = \frac{0.076}{0.9} = 0.08$ and $CR < 0.1$. Similarly to Table 2, we calculate all priority vectors of tasks, so the result is:

$$\Delta = \begin{pmatrix} 0.19 & 0.3 & 0.53 & 0.17 \\ 0.63 & 0.16 & 0.14 & 0.17 \\ 0.17 & 0.54 & 0.33 & 0.66 \end{pmatrix} \text{ and } \gamma = \begin{pmatrix} 0.09 \\ 0.5 \\ 0.28 \\ 0.12 \end{pmatrix} \text{ Thus, } PVM = \Delta.\gamma = \begin{pmatrix} 0.33 \\ 0.19 \\ 0.46 \end{pmatrix}$$

Table 1. Priority of AHP criterion

Criterion	Length	Waiting time	Burst time	Deadline	Priority vector
Length	1	1/3	1/4	1/2	0.09
Waiting time	3	1	3	4	0.48
Burst time	4	1/3	1	3	0.27
Deadline	2	1/4	1/3	1	0.14

Table 2. Priority of tasks according to the length criteria

Length	Task 1	Task 2	Task 3	Priority vector
Task 1	1.00	0.33	1.00	0.19
Task 2	3.00	1.00	4.00	0.63
Task 3	1.00	0.25	1.00	0.17

This means that task 3 has highest priority then task 1 has second priority and task 2 has less priority. So, the order of tasks in the global priority-queue will be: Task 3 in the front of queue, then task 1, next task 2.

4.2 Simulation Results

In the simulation experiments, we compare the proposed algorithm DPQ-PSO with FCFS, and PSO in terms of makespan. Several experiments with different parameter setting were performed to evaluate the efficiency of proposed algorithm.

Figure 3 shows the comparative analysis of makespan of the three algorithms. The makespan of DPQ-PSO is better when compared with the other two algorithms, because the tasks are prioritized into three different levels, then these tasks are submitted according to the priorities. High priority tasks have been submitted first and processed by VMs with high priority and so on. Therefore, high priority tasks take less time, then medium priority task, and then low priority tasks. DPQ-PSO can achieve an advantage performance and good utilization of resources.

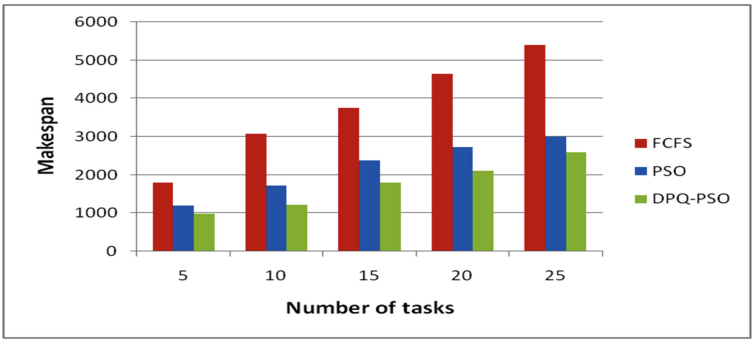


Fig. 3. Average makespan with different number of tasks

5 Conclusion

Scheduling is an interesting challenge in cloud computing nowadays. Priority of tasks is an important issue in task scheduling. This paper proposes an efficient Dynamic Priority-Queue for task scheduling in cloud computing based on multiple attribute decision making model using analytical hierarchy process to prioritize tasks, and also using meta-heuristic algorithm to schedule tasks to the best suitable resources. The simulation results show that the proposed approach acts successfully with good performances in term of makespan, achieve a high utilization of resources and also improve users' comprehensive QoS significantly. In future, the proposed work can be enhanced so as adding more QoS parameters.

References

1. Mell, P., Grance, T.: The NIST Definition of Cloud Computing. National Institute of Standards and Technology, the NIST Special Publication 800-145. ACM (2011)
2. Hoang, H.N., Le Van, S., Maue, H.N., Bien, C.P.N.: Admission control and scheduling algorithms based on ACO and PSO heuristic for optimizing cost in cloud computing. In: Król, D., Madeyski, L., Nguyen, N.T. (eds.) Recent Developments in Intelligent Information and Database Systems. SCI, vol. 642, pp. 15–28. Springer, Heidelberg (2016). doi:10.1007/978-3-319-31277-4_2
3. Thomas, A., Krishnalal, G., Jagathy Raj, V.: Credit based scheduling algorithm in cloud computing environment. Procedia Comput. Sci. **46**, 913–920 (2015)
4. Wu, X., Deng, M., Zhang, R., Zeng, B., Zhou, S.: A task scheduling algorithm based on QoS-driven in cloud computing. Procedia Comput. Sci. **17**, 1162–1169 (2013)
5. Karthick, A., Ramaraj, E., Subramanian, R.: An efficient multi queue job scheduling for cloud computing. In: 2014 World Congress on Computing and Communication (2014)
6. Verma, A., Kaushal, S.: Bi-criteria priority based particle swarm optimization workflow scheduling algorithm for cloud. In: RAECS (2014)
7. Patel, S., Bhoi, U.: Improved priority based job scheduling algorithm in cloud computing using iterative method. In: International Conference on Advances in Computing and Communications (2014)
8. Ghanbari, S., Othman, M.: A priority based job scheduling algorithm in cloud computing. Procedia Eng. **50**, 778–785 (2012)
9. Gu, L., Tang, Z., Xie, G.: The implementation of MapReduce scheduling algorithm based on priority. In: Li, K., Xiao, Z., Wang, Y., Du, J., Li, K. (eds.) ParCFD 2013. CCIS, vol. 405, pp. 100–111. Springer, Heidelberg (2014). doi:10.1007/978-3-642-53962-6_9
10. Xu, L., Yang, J.-B.: Introduction to Multi-Criteria Decision Making and the Evidential Reasoning Approach. Working Paper No. 0106, May 2001
11. Saaty, T.L.: The Analytic Hierarchy Process. McGraw-Hill, New York (1980)
12. Saaty, T.L.: Decision making with the analytic hierarchy process. Int. J. Serv. Sci. **1**(1), 83–98 (2008)
13. Alonso, J.A., Lamata, M.T.: Consistency in the analytic hierarchy process: a new approach. Int. J. Uncertain. Fuzziness Knowl.-Based Syst. **14**, 445–459 (2006)
14. Kennedy, J., Eberhart, R.: Particle swarm optimization. In: International Conference on Neural Networks, vol. 4, pp. 1942–1948. IEEE (1995)

15. Clerc, M., Kennedy, J.: The particle swarm – explosion, stability, and convergence in a multidimensional complex space. IEEE Trans. Evol. Comput. **6**, 58–73 (2002)
16. Feng, Y., Teng, G., Wang, A., Yao, Y.: Chaotic inertia weight in particle swarm optimization. In: Second ICICIC, p. 475. IEEE (2007)
17. Xin, J., Chen, G., Hai, Y.: A particle swarm optimizer with multi-stage linearly-decreasing inertia weight. In: International Joint Conference on Computational Sciences and Optimization, pp. 505–508. IEEE (2009)
18. Yue-lin, G., Yu-hong, D.: A new particle swarm optimization algorithm with random inertia weight and evolution strategy. In: International Conference on Computational Intelligence and Security (CISW 2007), pp. 199–203. IEEE (2007)
19. Kennedy, J., Eberhart, R.: A discrete binary version of the particle swarm algorithm. In: International Conference on Systems, Man, and Cybernetics. Computational Cybernetics and Simulation, pp. 4104–4108. IEEE (1997)
20. Saaty, T.: How to make a decision: the analytic hierarchy process. Eur. J. Oper. Res. **48**, 9–26 (1990)
21. Maysum, P.: Iterative methods for computing eigenvalues and eigenvectors. Waterloo Math. Rev. **1**, 9–18 (2011)
22. Calheiros, R., Ranjan, R., Beloglazov, A., De Rose, C., Buyya, R.: CloudSim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms. J. Softw.—Pract. Exp. **41**, 23–50 (2011). ACM
23. Parallel Workloads Archive: The Cornell Theory Center (CTC) IBM. http://www.cs.huji.ac.il/labs/parallel/workload/l_ctc_sp2/index.html

Hybridization of an Index Based on Concept Lattice with a Terminology Extraction Model for Semantic Information Retrieval Guided by WordNet

Fethi Fkih(✉) and Mohamed Nazih Omri

MARS Research Unit, Department of Computer Sciences,
Faculty of Sciences of Monastir, University of Monastir, Monastir, Tunisia
fethi.fkih@gmail.com
http://www.mars.org.tn/

Abstract. In this paper, we present IRAFCA-WN an information retrieval system from textual documents. Our model includes an automatic extraction module of terminology and a module based on formal concept analysis (FCA) guided by WordNet for indexing. The experimental study on a standard test data shows the effectiveness of our hybrid model.

1 Introduction

The design of an effective model for information retrieval requires three main factors. The first one is to have a reliable system for the extraction of meaningful terms that can describe, with a comprehensive and discriminative way, the content of the documents. The second important factor is to use a flexible indexing model which facilitates the organization of terms and documents. The third factor is the use of linguistic resources (lexical databases, thesauri, ontologies, etc.) to give a semantic aspect to the system and increase its performance [17,19].

Our system IRAFCA-WN tried to satisfy these requirements. First, we developed a terminology extraction system. Then we proposed an indexing model. Finally, we used WordNet[1] to solve problems related to the ambiguity of natural languages. The remainder of this paper is structured as follows. Section 2 presents the main approaches of Information retrieval based on concept lattice. In Sect. 3, we introduce our approach for the information retrieval based on concept lattice guided by WordNet. Section 4 is reserved for the performance tests of our approach. Our experimental study is realized on the standard test data Medline and our model is compared with other powerful models.

[1] https://wordnet.princeton.edu/wordnet/download/.

A. Abraham et al. (eds.), *Proceedings of the 16th International Conference on Hybrid Intelligent Systems (HIS 2016)*, Advances in Intelligent Systems and Computing 552,
DOI 10.1007/978-3-319-52941-7_15

2 Related Works

Since the introduction of the theory of formal concept analysis (FCA) in the 80s by Wille [21], the use of concept lattices for information retrieval has become very common. However, information retrieval is seen as a direct and intuitive application of the FCA theory. In fact, there's a high correspondence between the theoretical foundations of FCA and information retrieval techniques [12]. Practically, if objects are replaced by documents and attributes by terms; the formal context becomes an inverted file (index) commonly used for indexing documents.

Rapidly several models of information retrieval based on FCA have emerged. We cite, CLAIR [5] which is based on query insertion through classification-based reasoning. BR-Explorer system retrieves objects by classifying the query in a concept lattice organizing the considered objects [16]. Concept Lattice-based Ranking (CLR) methods are essentially based on the idea of considering the query as a new entry to the formal context [12–15]. Carpineto and Romano in [4] present a detailed study on the application of formal context analysis for information retrieval.

Several studies show that information retrieval systems based on concept lattices are more efficient than those based on Boolean model [3,11]. Complexity remains the major limitation of the systems based on concept lattice. This complexity is mainly manifested in large contexts. To reduce the computational complexity for systems based on concept lattice, several solutions are proposed. We cite, among others, ZooM [18] and iceberg lattices [20].

3 IRAFCA-WN: An Overview

In this section, we present the general architecture of our information retrieval system. IRAFCA-WN is composed essentially of two modules (see Fig. 1):

- A terminology extraction system COTEM: Our model of indexation is based mainly on the use of complex terms (i.e. terms composed by at least two tokens). Indeed, the use of complex terms to index a documents collection decreases the problems of semantic ambiguity that affects the performances of information retrieval systems. In previous work [6–8,10] we have presented COTEM a model of extraction of complex terminology from text documents. This model showed an acceptable efficiency. COTEM focuses primarily on the extraction of complex terms, in order to use them for indexing. Indexing by complex terms is justified by their low semantic ambiguities, which will greatly simplify the task of disambiguation what will save us time and resources. As we already mentioned in [6], COTEM consists of two modules that work sequentially (see Fig. 1): first, a linguistic extraction based on CRFs, then a statistical filter.
- An indexing architecture based on concept lattice model IRAFCA [9]: In order to organize the relationships between documents and terms, we exploit the theoretical basis provided by the formal concept analysis (FCA). In fact, using

a concept lattice for indexing documents has led us to propose IRAFCA, an information retrieval algorithm in a concept lattice. To give a semantic aspect to our index, we used a lexical database (WordNet) to manage problems associated with polysemy and synonymy which are very common in natural languages. The use of complex terms for indexing will greatly facilitate the task of the disambiguation. Indeed, a study of the WordNet2.0 ontology has shown that 89% of complex terms are monosemic (having only a single meaning) and 99% of complex terms have at most two different senses, unlike simple terms (composed by a single token) that are usually ambiguous [1]. For weighting the terms of our index, we use the weighting normalized function used in the Inquery system [2].

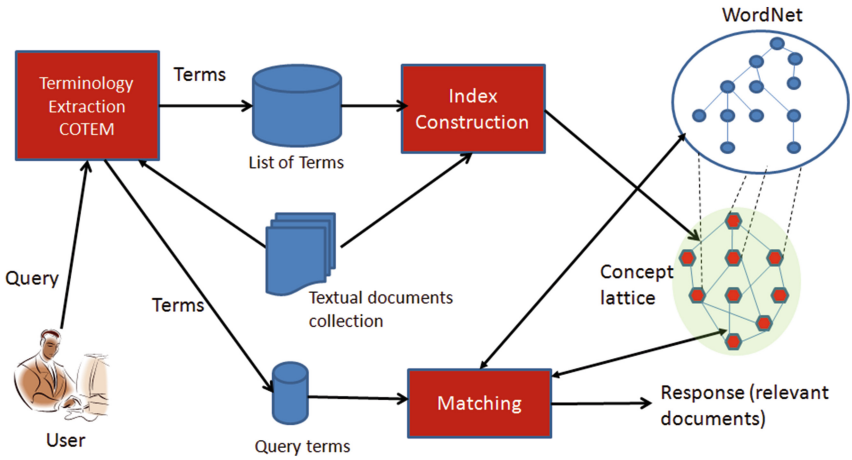


Fig. 1. General overview of IRAFCA-WN

3.1 Index Construction

In this level, we try to determine the concepts that correspond to complex terms extracted by COTEM. For this, we project these terms onto WordNet. Practically, these terms are classified into two categories:

1. Terms that are entries in WordNet: these terms do not present problems essentially if they are mono-sense. Otherwise, a disambiguation task becomes mandatory if they have several meanings (a small percentage according to statistics).
2. Terms that are not entries in Wordnet: we look for the closest concept such that its definition contains the maximum word of the considered term. After identifying the concepts, it remains to weight the relation between the concept and the document that provide it. This weight is calculated by quantifying the statistical and semantic importance of the concept in the document.

Our next step, is to construct the index. For this, we propose a model based on concept lattice. The use of the lattice provides a semantic model, since the index is the result of unsupervised classification operation (clustering). Algorithm 1 shows the steps required to build the index. Note that we use the ConExp[2] software to build a XML file corresponding to the lattice.

Algorithm 1. Index_Construction()

Data: $T = \{t_1, t_2, ..., t_n\}$: List of terms belonging to the index, with n is the number of terms in the index
WordNet: an ontology
Result: $B(D, T, I)$: Concept lattice modelling the relation between terms and documents

```
begin
1   | Disambiguation_Index()
    | for t ∈ T do
2   |   | Weight(t)
    | end
    | Build_Lattice()
end
```

3.2 Index Disambiguation

Our technique used for disambiguation (see Algorithm 2) is based on the set of terms mono-sense of the index. For each term having several possible meanings, we look for the closest sense to the set of concept already identified. Once the term is disambiguated it is inserted into the set. We repeat these tasks until completing all the terms.

3.3 Query Matching and Documents Ranking

For each user query, we apply COTEM to extract relevant terms. For each extracted term, we look (in the index) for the concept that can presenting it. If the index does not contain this term then we look for the concepts of the index that are closest (semantically) to the concepts of the query. After identifying documents corresponding to the request, they are classified in descending order of relevance. The degree of relevance is calculated based on statistical and semantic importance of query terms in the documents returned by the system. Algorithm 3 summarizes the instructions needed for the task of matching.

4 Comparative Evaluation

In this section we try to demonstrate that our method is more efficient than other methods in the literature. For this, we compared IRAFCA-WN with: BR-Explorer [16], CLAIR [5] and CLR [3].

[2] http://sourceforge.net/projects/conexp/.

Algorithm 2. Disambiguation_Index()

Data: $T = \{t_1, t_2, ..., t_n\}$: Set of terms belonging to the index, with n is the number of terms in the index
WordNet: an ontology
$t = \{c\#1, c\#2, .., c\#m\}$: set of possible sense of t in WordNet
MS: Set of mono-sense terms
Result: $CI = \{c_1, c_2, ..., c_n\}$: Identified concepts
begin
1 $CI \longleftarrow MS$
for $t \in T$ *and* $t \notin MS$ **do**
2 Find c such that $Distance(c, MS) = Max(Distance(c\#i, MS))_{i=1..m}$
3 $CI \longleftarrow CI \cup \{c\}$
end
$Build\_Lattice()$
end

Algorithm 3. Matching()

Data: $Q = \{t_1, t_2, ..., t_q\}$: Set of query terms
$T = \{t_1, t_2, ..., t_n\}$: terms of the index
$B(D, T, I)$: index
Result: $D_Q = \{d_1, d_2, ..., d_m\}$: Set of documents satisfying Q
begin
for $t \in Q$ **do**
1 **if** $t \notin T$ **then**
2 Disambiguate(t)
3 Find $C_{index} \in B$ the most similar concept to t
4 Weight(t)
end
end
5 $IRAFCA()$
end

4.1 Test Data Description

During this work, we used an extract (50 abstracts) from the medical collection Medline[3]. We start with cleaning the corpus which is a necessary task in order to remove all empty (stop) words such as articles (*the*, *a*, *an*, *some*, *any*), conjunctions (*before*, *when*, *so*, *if*, etc.), pronouns (*personal*, *relative*, *possessive* and *demonstrative*) and punctuation. Indeed, these words have high frequencies in documents without having a real terminology importance. We note that the lemmatization task is performed by TreeTagger[4].

For each abstract, we apply COTEM an automatic extraction system of complex terminology. To see the theoretical foundations of the terminology

[3] http://ir.dcs.gla.ac.uk/resources/test_collections/medl/.
[4] http://www.ims.uni-stuttgart.de/projekte/corplex/TreeTagger/.

extraction system (COTEM), you could see our previous articles [6,10]. Finally, we get a set of descriptors (terms) for each document. In total, we get 246 terms for the entire collection (see Table 1).

Table 1. Corpus description

Medline	
Documents number	Terms number
50	246

By identifying the terms that represent each document, we can build the formal context corresponding to the Medline corpus. Table 2 shows the lattice corresponding to the formal context used in our experimental study.

Table 2. Lattice description

	Medline lattice		
Attributes number	Objects number	Concepts number	Relations number
246	50	147	402

4.2 Experimentation and Results

To show the effectiveness of our system, we used a well-known technique in the information retrieval field. This technique is called "11-point interpolated average precision" that evaluates the performance of the ranking of documents returned by the system. Thus, this technique gives an idea about the relationship between precision and recall. For this purpose, we compare the results provided by our system with 3 other information retrieval models, namely: BR-Explorer, CLAIR and CLR. Table 3 and Fig. 2 summarizes the results (average precision) of each model.

4.3 Results Evaluation

To evaluate the curve provided by the "11-point average precision" method, we divide the result into two parts: the documents provided at the beginning of the list and the documents provided at the end of the list.

Regarding the documents provided in the top of the list, we find that IRAFCA-WN and CLAIR models are better than others, although CLAIR has a slight edge. This result is very important because, practically, the user focuses on the documents at the beginning of the list and ignores the rest.

Regarding the documents provided at the end of the list we find an advantage for IRAFCA-WN on other models. The Fig. 2 shows that there are no great

Table 3. 11-point interpolated average precision for each model.

Recall	IRAFCA-WN	BR-Explorer	CLAIR	CLR
0	0.7356	0.6142	0.7671	0.6429
0.1	0.6924	0.6015	0.7214	0.6177
0.2	0.6511	0.5672	0.701	0.5149
0.3	0.593	0.5436	0.6344	0.5098
0.4	0.5732	0.5181	0.6031	0.4252
0.5	0.5493	0.4913	0.5174	0.4007
0.6	0.5327	0.4751	0.4125	0.3262
0.7	0.518	0.4423	0.4027	0.3127
0.8	0.4571	0.4168	0.33685	0.3056
0.9	0.4488	0.3573	0.3019	0.2714
1	0.412	0.3292	0.2598	0.2352

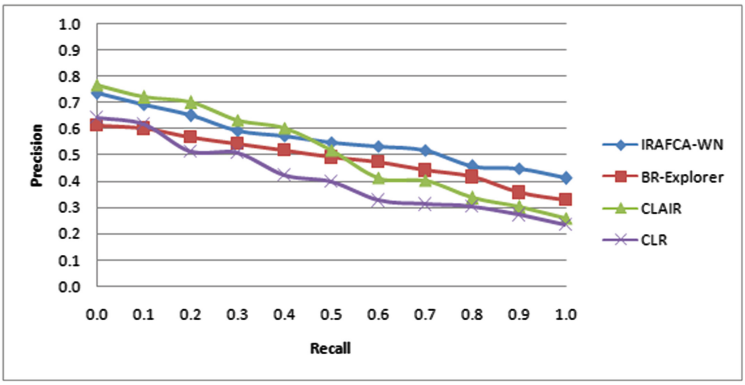


Fig. 2. 11-point interpolated average precision

Table 4. Average precision for each model.

Model	Average precision
IRAFCA-WN	0.56029091
CLAIR	0.51437727
BR-Explorer	0.48696364
CLR	0.41475455

variations in the precision curve and there is no sudden drop. Indeed, IRAFCA-WN starts with 0.7 and ends with the value 0.4.

An overview of the results can be drawn from the Table 4 and the Fig. 3. Indeed, the IRAFCA-WN model gives a good average precision and it is clearly

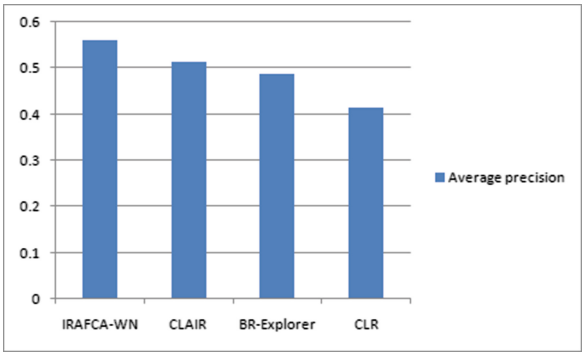


Fig. 3. Average precision

better to other models. This can be explained by the use of WordNet which decreased issues related to the phenomenon of polysemy which improved system accuracy.

5 Conclusion

In this paper, we presented IRAFCA-WN a system for indexing and information retrieval. Our model includes a module for automatic term extraction from textual documents, a matching module between query and index and an indexing model based on concept lattice guided by WordNet.

We conducted a comparative study between IRAFCA-WN with other models in the literature. This study showed the good performance of our system and its superiority to other systems.

As future work, we will try to improve the terminology extraction module by integrating deeper linguistic knowledge.

References

1. Baziz, M.: Indexation conceptuelle guide par ontologie pour la recherche d'information. Ph.D. thesis, Universit Paul Sabatier (2005)
2. Callan, J., Croft, W.B., Harding, S.M.: The INQUERY retrieval system. In: Proceedings of the Third International Conference on Database and Expert Systems Applications, pp. 78–83. Springer (1992)
3. Carpineto, C., Romano, G.: Order-theoretical ranking. J. Am. Soc. Inf. Sci. **51**(7), 587–601 (2000)
4. Carpineto, C., Romano, G.: Concept Data Analysis: Theory and Applications. Wiley, Chichester (2004)
5. Codocedo, V., Lykourentzou, I., Napoli, A.: A semantic approach to concept lattice-based information retrieval. Ann. Math. Artif. Intell. **72**(1–2), 169–195 (2014)

6. Fkih, F., Omri, M.N.: Complex terminology extraction model from unstructured web text based linguistic and statistical knowledge. IJIRR **2**(3), 1–18 (2012)
7. Fkih, F., Omri, M.N.: Estimation of a priori decision threshold for collocations extraction: an empirical study. IJITWE **8**(3), 34–49 (2013)
8. Fkih, F., Omri, M.N.: A statistical classifier based Markov chain for complex terms filtration. In: Proceedings of the International Conference on Web Informations and Technologies, ICWIT 2013, pp. 175–184. Hammamet, Tunisia (2013)
9. Fkih, F., Omri, M.N.: IRAFCA: an O(n) information retrieval algorithm based on formal concept analysis. Knowl. Inf. Syst. **48**(2), 465–491 (2016)
10. Fkih, F., Omri, M.N., Toumia, I.: A linguistic model for terminology extraction based conditional random field. In: Proceedings of the International Conference on Computer Related Knowledge, ICCRK 2012, Sousse, Tunisia, p. 38 (2012)
11. Godin, R., Mineau, R., Missaoui, R., Mili, H.: Méthodes de classification conceptuelle basées sur les treillis de galois et applications. Revue d'intelligence artificielle **9**(2), 105–137 (1995)
12. Godin, R., Missaoui, R., Alaoui, H.: Incremental concept formation algorithms based on Galois (concept) lattices. Comput. Intell. **11**(2), 246–267 (1995)
13. Kourie, D.G., Obiedkov, S., Watson, B.W., van der Merwe, D.: An incremental algorithm to construct a lattice of set intersections. Sci. Comput. Program. **74**(3), 128–142 (2009)
14. La, P.-T., Le, B., Vo, B.: Incrementally building frequent closed itemset lattice. Expert Syst. Appl. **41**(6), 2703–2712 (2014)
15. Merwe, D., Obiedkov, S., Kourie, D.: AddIntent: a new incremental algorithm for constructing concept lattices. In: Eklund, P. (ed.) ICFCA 2004. LNCS (LNAI), vol. 2961, pp. 372–385. Springer, Heidelberg (2004). doi:10.1007/978-3-540-24651-0_31
16. Messai, N., Devignes, M.-D., Napoli, A., Smaïl-Tabbone, M.: BR-Explorer: an FCA-based algorithm for information retrieval. In: Fourth International Conference on Concept Lattices and Their Applications - CLA 2006, Hammamet/Tunisia, October 2006
17. Naouar, F., Hlaoua, L., Omri, M.N.: Collaborative information retrieval model based on fuzzy confidence network. J. Intell. Fuzzy Syst. **30**(4), 2119–2129 (2016)
18. Pernelle, N., Rousset, M.-C., Soldano, H., Ventos, V.: Zoom: a nested galois lattices-based system for conceptual clustering. J. Exp. Theor. Artif. Intell. **14**(2–3), 157–187 (2002)
19. Sendi, M., Omri, M.N.: Biomedical concept extraction based information retrieval model: application on the mesh. In: 2015 15th International Conference on Intelligent Systems Design and Applications (ISDA), pp. 40–45, December 2015
20. Stumme, G., Taouil, R., Bastide, Y., Lakhal, L.: Conceptual clustering with iceberg concept lattices. In: Proceedings of the GI-Fachgruppentreffen Maschinelles Lernen 2001, October 2001
21. Wille, R.: Restructuring lattice theory: an approach based on hierarchies of concepts. In: Ferré, S., Rudolph, S. (eds.) ICFCA 2009. LNCS (LNAI), vol. 5548, pp. 314–339. Springer, Heidelberg (2009). doi:10.1007/978-3-642-01815-2_23

On Integrating Simulated Annealing Within Parallel Genetic Algorithm: An On-Demand Transportation Case Application

Olfa Chebbi(✉), Ezzeddine Fatnassi, and Hadhami Kaabi

Institut Supérieur de Gestion de Tunis, Université de Tunis,
41, Rue de la Liberté, Bouchoucha, 2000 Bardo, Tunisia
olfaa.chebbi@gmail.com

Abstract. The increased use of private vehicles, especially in urban areas, has led to problems with pollution and congestion. In response, new transportation systems, such as Personal Rapid Transit (PRT), have been developed. In this paper, we propose to deal with a routing problem related to PRT that consists of satisfying a known set of passenger requests using a set of homogeneous PRT vehicles with limited battery capacity. Our primary goal is to test the effect of integrating simulated annealing algorithm within parallel genetic algorithm in this particular case of on-demand transportation system. For that purpose, this paper proposes an hybrid simulated annealing genetic algorithm for the case of energy minimization in PRT. In this paper, we demonstrate the efficiency of the proposed algorithm by testing it on a large number of PRT instances adapted from a real case study. Our method is shown to produce good results as we found an average gap relative to a linear relaxation of 2.131%.

Keywords: Personal Rapid Transit · Management of public transportation · Empty vehicle redistribution · Vehicle routing problem

1 Introduction

1.1 Background of the Paper

Nowadays, urban population is growing at an incredible pace [8]. Urban areas around the world are changing drastically. They are facing continuously an increasing number of challenges in term from efficient transportation of persons and goods. In fact, city stackeholders need to create an efficient urban mobility tools that contribute on reaching social, economic and environmental objective of urban areas while reducing its negative impact such as pollution, carbon emission, congestion, etc.

To do so, we focus in this paper on the Personal Rapid transit system (PRT) as a new innovative public transportation tool. The PRT is a public transportation mode that could be seen as an efficient, ecological, transportation system.

A. Abraham et al. (eds.), *Proceedings of the 16th International Conference on Hybrid Intelligent Systems (HIS 2016)*, Advances in Intelligent Systems and Computing 552,
DOI 10.1007/978-3-319-52941-7_16

It has the possibility to overcome the classical problem related to public transportation. PRT uses small driverless electric vehicles to offers an **on-demand taxi-like** transportation service for its users.

PRT have recently proved its efficiency and is now used in different urban contexts such as airports (Heathrow London UK), university campus (Morgan town, West Virginia USA), new ecological city (Masdar City Abu Dhabi UAE), etc.

1.2 Related PRT Literature

As the PRT is considered as a relatively new transportation system, its literature is at early stage. Despite early ideas related to PRT was presented in 1953 by Don Fichter, there is only about 200 papers that treated different PRT issues [6]. The related literature related to PRT have focused on the feasibility of such an innovative system and only recently more works have been focused on the operational level and issues of PRT. One could note specially the optimal design problem [17], optimizing the fleet size [4], simulation [11], theoretical maximum capacity [13], etc. More recently a special focus was made on different routing problems related to PRT. One could note specially the problem of minimizing the waiting time of passengers [7] and the minimization of energy consumption [10,14].

In this work, we consider to minimize the energy minimization of the PRT in a static deterministic context while considering the use of battery operated PRT vehicles. In fact, the recent works of minimizing energy minimization of the PRT [14,15] considered to develop exact and mat-heuristic methods. However investigating the hybridizing of different heuristics to solve such a problem was not investigated in the literature.

Therefore and starting from this gap in the literature, we consider hybridizing simulated annealing algorithm and parallel genetic algorithm in order to minimize energy consumption while focusing on the battery issues for RT vehicles. This is considered as a complication factor as vehicles would be periodically unavailable to recharge their batteries in the depot.

1.3 Motivation of the Studied Problem

This work is concerned with the problem of minimizing energy consumption for PRT system in a static deterministic context. More specifically, we focus on the empty vehicle management (EVM) of PRT vehicles under battery constraints to minimize the total energy used to satisfy a set of known list of passengers requests.

In fact and as the PRT system offers an on-demand transportation service, the number of vehicles ending at a specific station doesn't necessarily equal to the number of the vehicles departing from that station. Therefore, some PRT vehicles needs to move empty (without passengers) to cope with the upcoming demand at certain stations. The problem of deciding which specific empty vehicles to move and where to move it is called the EVM. Solving this problem is highly important to PRT system as it would ensure a maximum level of service using

the limited resource of PRT vehicle. This is highly important for a PRT system in order to be competitive against other transportation tools.

1.4 The Proposed Solution Approach

When solving an optimization problem such as the minimization of energy consumption of PRT, one could implement exact or meta-heuristic solution approaches. In fact, the quality of the generated solutions depends closely on the chosen solution approach and the problem size. When solving small instances, exact methods are most suited to solve them. As for large instances, exact methods are not adapted to solve them as they consume a large amount of computational time. On the other hand, approximate approach such as simulated annealing and genetic algorithms, are known to be a powerful method for generating high quality solutions in a small computational time in the transportation context.

Within this context, one could focus specially on parallel genetic algorithms. Parallel genetic algorithms are based on a conceptual model where parallel populations exist and evolve together in order to find the best solution for an optimization problem. Parallel genetic algorithms offers the advantages of exploiting new hardware architectures such as multi-cores computers in order to develop new kind of genetic algorithms that run more efficiently profiting from these architectures.

In this paper, we consider implementing a parallel genetic algorithm for solving the proposed problem based on the different advantages that such a model offer. Also, we consider a learning local search method such as simulated annealing to be implemented within the proposed parallel genetic algorithm. A simulated annealing (SA) is a stochastic algorithm that can be seen as a hill-climber which implement internal mechanism to escape from local optima. For our algorithm, moves that increase the energy consumption of the system would be accepted with a decreasing probability.

We should note that in the literature several parallel genetic algorithms were implemented to solve different problems [2,9,12]. However, the hybridization of SA with a parallel genetic algorithm was not considered in the transportation context before.

1.5 Contribution of This Paper

In this work, and in emphasis with the idea of implementing an efficient hybrid meta-heuristic for energy minimization in the context of PRT, we:

1. We focus on the operational level of decisions related to PRT and in order to enhance its performance, we propose to study a new problem related to PRT in order to minimize the energy minimization of the system in a static deterministic context while using vehicles with limited battery capacity.
2. We propose to hybridize a parallel genetic algorithm with SA in order to solve the proposed problem.

3. We validate our finding and model through extreme experimental study based on the case study of the Masdar PRT' network in the United Arab Emirates (UAE) taken from the literature.

1.6 Outline of This Paper

This paper is organized as follow: we first present the formal definition of our problem related to PRT. Section 3 presents the hybridization of SA and parallel genetic algorithm to deal with our problem. Section 4 proposes the experimental study to test the validity of our algorithm and the efficiency of introducing SA within parallel genetic algorithm. Finally, the conclusions are presented in Sect. 5.

2 Problem Definition

The problem treated is based on the assumption that we have a predetermined list of trips to serve. We define the problem as presented by Mrad and Hidri.

Let us suppose that we have a set of PRT stations S, a depot D, and a PRT network N that makes it possible to journey between any pair of stations. Let us also suppose that we have an unlimited number of vehicles that are initially located in the depot and have battery capacity B.

$Cost_{i,j}$ is a cost matrix that defines the cost of traveling from station i to station j. The cost of moving between station i and station j in the PRT network will be the length of the shortest path calculated using a Floyd–Warshall algorithm.

Let us define a list of trips T that has cardinality $|T| = n$. Each trip i $(= 1, \ldots, n)$ will have its particular origin and arrival stations (OS_i, AS_i) and its particular departure and arrival times (OT_i, AT_i). We also suppose that the vehicle should return to the depot to charge its battery when necessary. The PRT problem is defined on an asymmetric graph $G = \{\boldsymbol{V}, \boldsymbol{E}\}$, where $\boldsymbol{V} = \{v_0, v_1, v_2, \ldots, v_n\}$ is a set of nodes for which v_0 defines the depot and $v_1, v_2, \ldots, v_n$ defines the n different trips that the PRT system must cover. We define $\boldsymbol{V}^* = \boldsymbol{V}/v_0$. Moreover, $\boldsymbol{E} = \{(v_i, v_j); v_i, v_j \in \boldsymbol{V}\}$ is a set of arcs that is defined as follows:

- If $v_i, v_j \in V^*$ with $AT_i + Cost_{(AS_i, OS_j)} \leq OT_j$, then the arc (i, j) exists and has cost c_{ij}, which represents the energy consumed from the arrival station of trip i (AS_i) to the arrival station of trip j (AS_j). Hence, each edge will have a combined cost that includes the cost of the movement from one trip to another and the cost of a trip itself.
- For each node $i \in V^*$ we add an arc $(0, i)$. The cost of this arc is c_{0i} and represents the energy used to reach the arrival station of trip i from the depot.
- For each node $i \in V^*$ we add an arc $(i, 0)$. The cost of this arc is c_{i0} and represents the energy used to reach the depot from the arrival station of trip i.

We also define $\boldsymbol{E}' = \{(v_i, v_j); v_i, v_j \in \boldsymbol{V}^*\}$.

Note that G is a direct incomplete graph, because if the arc between node i and node j exists, the opposite arc does not exist. Our problem is a typical node routing problem that can be assimilated into the asymmetrical distance-constrained vehicle routing problem (ADCVRP) [5]. This problem is proven to be NP-hard [5]. The main purpose in our problem is to assign trips to vehicles with respect to the battery capacity of each vehicle in order to minimize the total consumption of electrical energy.

3 Hybridizing Simulated Annealing and Parallel Genetic Algorithm

In this section, we describe our parallel hybrid genetic algorithm (GA) and SA used to solve the energy minimization PRT problem. First, we present the main details about the proposed algorithm (solution representation, fitness evaluation, crossover and mutation operators). Then, we discuss the parallel model used.

3.1 Solution Encoding

We used the permutation representation with integer number encoding. A permutation of integer would represent the different trips of our problem. A solution based on our solution encoding would require for each trip an unique integer ID. For example, 5 trips' problem will need five different identifiers: 1, 2, 3, 4, 5. The permutation representation offers the possibility to use specific operators in order to make sure to obtain a feasible solutions.

3.2 Solution Evaluation

A fitness function is an important feature for any GA as it defines how good a solution is. Thus, fitness function guides the GA during its search process toward the optimal solution. In this paper, we adapt the Split fitness function of Prins used initially for the capacitated vehicle routing problem (CVRP) [16].

The Split function founds the optimal set of roads starting from a permutation by computing a shortest path in an auxiliary graph H. The Split function first constructs an auxiliary graph H where each node represents a trip in the solution and each edge represents a feasible road. H contains also an additional node representing the depot location. Therefore for a problem containing n trips, H would contain $n+1$ nodes numbered from 0 to n. Each edges in H has a cost representing the energy consumption relative to its related road. Let us recall that a road is a sequence of visiting trips starting and ending at the depot. To determine the related cost of a solution, the Split function computes the shortest path in H starting from node 0 to node n. More details about the Split function could be found in [16].

3.3 Genetic Operators

In order to generate a new solutions, GA applies two main operators: crossover and mutation. As for the crossover operator, we used in our algorithm the one point crossover. This operator and starting from the two selected parents, chooses a random cutting point. The trips that appear before the cutting point are copied into the new generated offspring. The missing trips in the offspring are copied from the second parent while respecting their order of appearance.

As for the mutation operator, we used the insertion mutation. A random trip is selected and is inserted in a new different position randomly.

3.4 The Parallel Approach

A parallel GA is characterized by having multiple component, which are independent of their the general population structure. In this paper, we choose the cellular parallel model.

The main advantage and characteristic of the cellular parallel model is the structure of its population. In fact, the population of individuals in a cellular parallel model are in the form of a matrix of individuals. Each individual in the matrix population interact only with its neighbors. Consequently and since good quality solutions could be present in different areas of the population, they are slowly spread through the global population.

Consequently, the proposed parallel solution approach is proposed in Algorithm 2. We should note that the proposed parallel solution approach is hybridized with an SA procedure presented in Algorithm 1. Note that the function C used in our Algorithms refers to the evaluation function.

4 Experimental Validation

In order to test the ideas presented in this paper and the effect of integrating SA into a parallel genetic algorithm in the context of PRT, we used instances from the literature [15]. These instances are based on the use case of Masdar City in the UAE. The size of the list of trips of these instances varies from 10 to 400 trips. Overall this testing bed has 1320 instances. To assert the quality of the obtained solutions, we used the GAP metric. The GAP is obtained as follows:

GAP $= (\frac{(SOL-LB)}{LB}) \times 100$ We should note that LB is the linear relaxation of the presented mathematical formulation presented in the literature [1].

Results are obtained in Table 1.

For the genetic algorithm, the parameters are as follow: (i) population size: 16 (4*4), (ii) $max_{generations}$: 1000, (iii) crossover rate: 0.9, (iv) mutation rate: 0.3. For the SA procedure the parameter are as follow: (i) $EL = 20$, (ii) $MTT = 20$, (iii) $\alpha = 0.95$, (iv) $T_0 = 5$.

We should note that these parameters are defined based on a specific tuning algorithm that was presented in [3].

Table 1 proves the good quality results obtained from hybridizing SA and parallel genetic algorithm as we found an average gap of 2.131% in 9.424 s. In order

Table 1. Result of the proposed algorithm

Number of trips	Average gap %	Average time (sec)
10	1.195	0.908
15	1.031	1.173
20	1.138	1.456
25	1.671	1.765
30	1.796	2.087
35	2.224	2.285
40	1.869	2.579
45	1.701	2.941
50	1.797	3.255
55	2.260	3.621
60	1.752	3.813
65	1.773	4.553
70	2.530	4.637
75	2.192	5.044
80	2.017	5.187
85	2.250	5.768
90	1.899	6.187
95	2.176	6.428
100	2.452	6.850
110	2.059	7.597
120	2.383	8.469
130	2.748	9.147
140	1.987	9.811
150	2.004	10.802
160	2.654	12.065
170	2.350	12.592
180	2.038	13.682
190	2.392	15.176
200	2.711	15.934
250	2.267	21.277
300	2.672	27.318
350	2.919	34.883
400	3.402	41.698
Average	2.131	9.424

Algorithm 1. SA Procedure (Initial Solution:X_0)

```
r = 0
X_best = X_0
while (R ≤ MT)AND(T_r ≥ 0) do
  n = 0
  while n ≤ EL do
    Select Operator Exchange-Mutation, Inversion-Mutation or Displacement-Mutation Randomly
    X_new = X_n
    Δ = C(X_new) − C(X_best)
    if Δ < 0 then
      X_best = X_new
      X_n = X_new
      n=n+1
    else
      y=RAND(0, 1)
      Z = e^(−Δ/T)
      if y < Z then
        n=n+1
        X_n = X_new
      end if
    end if
    n=n+1
  end while
  r=r+1
  T_r = T_{r−1} − αT_{r−1}
end while
```

to test the effect of integrating SA within our proposed parallel algorithm, we performed a lesion study where we eliminate the SA component of our proposed algorithm and test the new algorithm of the instances of size great or equal to 200 trips. The difference on the performance of the two algorithms would provide us an idea about the expanse of the SA component on the performance of the proposed algorithm.

Also and to enrich our comparative study, we used the Wilcoxon matched-pairs signed-rank test in order to know if the results are statistically different. These comparative analysis are made on large instances of size great or equal to 200 trips. Results of this statistical test found a P-value less than 0.0001 which proves that the results of the hybrid SA and parallel genetic algorithm and the results of the basic parallel genetic algorithm are statistically different at the level of 0.01. These results confirm the good effect of integrating SA within a parallel model. In fact, the SA helps the parallel genetic algorithm to gain an average gap of 4.558% on instances of size big or equal to 200 trips. These results confirm the validity of integrating SA within a parallel algorithm.

Algorithm 2. Cellular Genetic Algorithm (List of Trips T, PRT network: N)

```
Constructs G based on T and N
while Not reach termination criterion do
  for x = 1 → WIDTH do
    for y = 1 → HEIGHT do
      parent1 = Individual-At(x,y).
      parent2 = Select-From-Neighborhood(x,y).
      offspring = One Point Crossover(parent1,parent2).
      offspring = Insertion Mutation(offspring).
      Evaluate The generated offspring
      Δ = C(offspring) − C(Individual − At(x, y))
      if offspring is better than Individual-At(x,y) then
        Insert(offspring, popaux, x,y).
      else
        Insert(Individual − At(x, y), popaux, x,y).
      end if
    end for
  end for
  pop = popaux
  for x = 1 → WIDTH do
    for y = 1 → HEIGHT do
      Apply SA on Individual-At(x,y)
    end for
  end for
end while
return the best individual in pop
```

5 Conclusions

In this paper, we studied the energy minimization for driverless vehicles in the context of PRT under time window and battery constraints. Due to the complexity of the problem under study, we have proposed an efficient hybrid SA and parallel genetic algorithm to minimize energy consumption. To achieve a robust design of the proposed algorithm, a cellular parallel model was proposed. Furthermore, computational experiments were carried out through 1320 different problem instances. Based on the obtained results, it was emerged that the proposed algorithm is able to obtain good quality result within a reasonable amount of CPU time for most cases. In addition, we can see that when the size of problems increase, the proposed hybrid algorithm still has a good performance in terms of solution quality.

Future research may focus on hybridizing parallel genetic algorithm with other learning individual techniques. Also other parallel models for genetic algorithms could be investigated. Also, using multi-objective models in which conflicting objectives are considered can be suggested for further research.

References

1. Almoustafa, S., Hanafi, S., Mladenovi, N.: New exact method for large asymmetric distance-constrained vehicle routing problem. Eur. J. Oper. Res. **226**, 386–394 (2012)
2. Cai, P., Cai, Y., Chandrasekaran, I., Zheng, J.: Parallel genetic algorithm based automatic path planning for crane lifting in complex environments. Autom. Constr. **62**, 133–147 (2016). http://www.sciencedirect.com/science/article/pii/S0926580515002009
3. Chebbi, O., Chaouachi, J.: Effective parameter tuning for genetic algorithm to solve a real world transportation problem. In: 2015 20th International Conference on Methods and Models in Automation and Robotics (MMAR), pp. 370–375. IEEE (2015)
4. Chebbi, O., Chaouachi, J.: Optimal fleet sizing of personal rapid transit system. In: Saeed, K., Homenda, W. (eds.) CISIM 2015. LNCS, vol. 9339, pp. 327–338. Springer, Heidelberg (2015). doi:10.1007/978-3-319-24369-6_27
5. Chebbi, O., Chaouachi, J.: Reducing the wasted transportation capacity of personal rapid transit systems: an integrated model and multi-objective optimization approach. Transp. Res. Part E Logistics Transp. Rev. (2015)
6. Cottrell, W.D.: Critical Review of the Personal Rapid Transit Literature, pp. 1–14 (2005). Chap. 34
7. Daszczuk, W.B., Choromański, W., Mieścicki, J., Grabski, W.: Empty vehicles management as a method for reducing passenger waiting time in personal rapid transit networks. IET Intell. Transp. Syst. **9**(3), 231–239 (2014)
8. Davis, K.: The urbanization of the human population. In: The City Reader, pp. 2–11 (2011)
9. El-Alfy, E.S.M., Alshammari, M.A.: Towards scalable rough set based attribute subset selection for intrusion detection using parallel genetic algorithm in mapreduce. Simul. Model. Pract. Theory **64**, 18–29 (2016). http://www.sciencedirect.com/science/article/pii/S1569190X16000174. Advances on Information and Communication Systems
10. Fatnassi, E., Chebbi, O., Chaouachi, J.: Discrete honeybee mating optimization algorithm for the routing of battery-operated automated guidance electric vehicles in personal rapid transit systems. Swarm Evol. Comput. **26**, 35–49 (2015)
11. Fatnassi, E., Chebbi, O., Siala, J.C.: Two strategies for real time empty vehicle redistribution for the personal rapid transit system. In: 2013 16th International IEEE Conference on Intelligent Transportation Systems-(ITSC), pp. 1888–1893. IEEE (2013)
12. He, L., Yu, X., Huang, Z., Talab, A.M.A.: The two-dimensional double-entropy threshold based on the parallel genetic simulated annealing algorithms. Optik - Int. J. Light Electron Opt. **127**(1), 96–101 (2016). http://www.sciencedirect.com/science/article/pii/S0030402615012759
13. Lees-Miller, J.D., Hammersley, J.C., Wilson, R.E.: Theoretical maximum capacity as benchmark for empty vehicle redistribution in personal rapid transit. Transp. Res. Rec. J. Transp. Res. Board **2146**(1), 76–83 (2010)
14. Mahdi, M., Lotfi, H.: Optimal consumed electric energy for a personal rapid transition transportation system. Comput. Indus. Eng. **79**, 1–9 (2015)
15. Mrad, M., Chebbi, O., Labidi, M., Louly, M.A.: Synchronous routing for personal rapid transit pods. J. Appl. Math. **2014**(1), 1–8 (2014)

16. Prins, C.: A simple and effective evolutionary algorithm for the vehicle routing problem. Comput. Oper. Res. **31**(12), 1985–2002 (2004)
17. Won, J.M., Choe, H., Karray, F.: Optimal design of personal rapid transit. In: 2006 IEEE Intelligent Transportation Systems Conference, ITSC 2006, pp. 1489–1494. IEEE (2006)

A Survey of Methods and Performances for EEG-Based Emotion Recognition

Asma Baghdadi[1(✉)], Yassine Aribi[1,2], and Adel M. Alimi[1,2]

[1] REGIM: REsearch Groups in Intelligent Machines, Sfax, Tunisia
{baghdadi.asma.tn,yassine.aribi.tn,adel.alimi}@ieee.org
[2] University of Sfax, National School of Engineers, BP 1173, 3038 Sfax, Tunisia

Abstract. EEG-based Emotion Recognition is regarded as a new field of affective computing researching, though presenting many challenging issues concerning the manner how emotions are elicited, and the different techniques used for features extraction and their ability to achieve high classification performance. This article reviews the Emotion Recognition techniques applied and developed recently. In general terms, emotion evocation based on audio-visual stimuli, features extraction techniques and classifiers are surveyed in the field of Emotion Recognition. A comparative table of recent researches is also conducted. Based on a discussion of previous studies, our proposed architecture is presented as future work.

Keywords: Electroencephalogram (EEG) · Emotion recognition · Performance · Fractal Dimension (FD) · Fuzzy C-Means (FCM)

1 Introduction

During the past twenty years, computer modeling of emotions has become an increasingly recognized thematic. Many researches in neurophysiology and neuropsychology were able to establish a strong link between emotion, rationality and decision making, having increased the interest of inclusion emotions in human interactions. Thus, different researchers in various fields, such as affective computing, became interested in the emotional dimensions.

Psychologists, physiologists, anthropologists, sociologists, philosophers and ethnologists have all studied different emotions in different contexts. Lately, the emotional component was significantly considered and developed in technological contexts such as robotics, human-computer interaction, and virtual reality.

In the context of assistive technologies, affective computing is an ongoing research domain that is highly associated with psychophysiology. Emotions reflect the inner senses of a person. Which is why affect detection systems are considered highly important in different cases like mental disorders analysis.

This article presents various preliminaries such as EEG signal bands and Stimuli datasets. It outlines the Emotion Recognition process specifying the different techniques and methods used in each step. A performance comparative table of researches is presented. Finally, based on a discussion of previous studies, our proposed architecture is presented as future work.

A. Abraham et al. (eds.), *Proceedings of the 16th International Conference on Hybrid Intelligent Systems (HIS 2016)*, Advances in Intelligent Systems and Computing 552,
DOI 10.1007/978-3-319-52941-7_17

2 Preliminaries

In the literature, many models, techniques and datasets are marked as references to be used in future researches.

This section describes different notations and theories referenced in this paper, such as the Russell's model for emotion mapping and datasets of labeled pictures and sounds.

2.1 EEG Signal Bands

The EEG is an electrical signal that is recorded by electrodes appropriately placed on the scalp of the head. These waves are generated from the brain, then amplified and displayed using a computer, or other suitable instrument. It consists of a wave that varies in time, much like a sound or speech signal. Frequency components of these signals can be measured and analysed, and relevant proprieties can be extracted after applying a frequency transform.

Based on an EEG signal, we are able to identify the subject's perspective state, so we can say "the subject is thinking about something exciting" or "the subject is thinking about something sad". However, it cannot inform us about the exact nature of the thought.

Derived properties from an EEG signal are found useful. Many EEG-based system are developed recently, such as Automatic Epilepsy Detection and Emotion Recognition.

Table 1. Summary of EEG frequency bands [32]

Band	Distribution	Feeling states	Tasks & Behaviors	Physiological correlates
Delta (0.1–3 Hz)	Broad or diffused, bilateral, widespread	Deep, dreamless sleep, non-REM sleep, trance, unconscious	Lethargic, not moving, not attentive	Not moving, low-level of arousal
Theta (4–7 Hz)	Regional, involve many lobes, lateralized or diffused	Intuitive, creative, recall, fantasy, imagery, creative, dreamlike, switching thoughts, drowsy	Creative, intuitive; distracted, unfocused	Healing, integration of mind/body
Alpha (8–12 Hz)	Regional, involves entire lobe; strong occipital w/eyes closed	Relaxed, not agitated, but not drowsy; tranquil, conscious	Meditation, no action	Relaxed, healing
Low beta (12–15 Hz)	Localized by side and by lobe	Relaxed yet focused, integrated	Resting yet alert	Sensorimotor rhythm
Midrange beta (15–18 Hz)	Localized, over various areas. May be focused on one electrode	Thinking, aware of self & Surroundings	Mental activity	Alert, active, but not agitated
High beta (above 18 Hz)	Localized, may be very focused	Alertness, agitation	Mental activity	General activation of mind & Body functions
Gamma (40 Hz)	Very localized	Thinking; integrated thought	High-level information processing	Information-rich task processing

Table 1 summarize briefly the basic EEG rhythms, considering their typical distribution on the scalp, feeling states, behaviors and physiological correlates.

2.2 Stimuli Datasets

The International Affective Picture System (IAPS) along with the International Affective Digitized Sound system (IADS), as well as other collections of affective stimuli, are Datasets developed specifically to provide a set of normative emotional stimuli for experimental investigations and researches of emotion and attention [23].

The objective is to simply develop a large set of standardized, emotionally-evocative, internationally accessible, colored photographs that includes contents across a wide range of semantic categories.

The IAPS disposes of 1200 pictures divided into 20 sets in which, each set consists of 60 photos. Each picture is labeled with valence and arousal values.

Several studies [7,9,10], have used pictures from IAPS to evoke subject emotions during exposition sessions. Others used extracted sets from IADS in order to record EEG data from volunteers [1].

2.3 The Russell's Model: Valence-Arousal Model

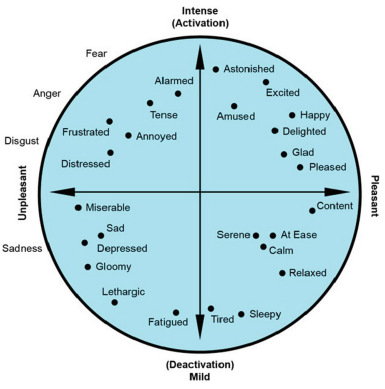


Fig. 1. The Russell's model of affect [31]

The continuous emotion Model of Russell disposes of two Primary dimensions, Valence for emotion positivity/negativity and Arousal for the excitement level. The secondary dimension is called Dominance or Control. This dimension is not necessary to determine an emotion [31].

Each emotion is a projection of valence and arousal levels such as excited, sad, happy and calmness. Several researches are based on the Russell circumplex model to recognize emotions (Fig. 1).

Whereas, researches do not achieve the complex task to recognizing all emotions because some of them are very similar (i.e. Happy and delighted). Which is why some researchers are limited to determine the main emotions such as happy, angry, sad, calm and neutral.

3 Emotion Recognition Process

EEG-based brain-computer interfaces are comprised of typical components; each element executes a specific critical function. The whole process is presented in Fig. 2.

3.1 Stimulus

It is essential to design efficient and reliable emotion elicitation stimuli for emotion experiments. Nowadays, there are numerous kinds of stimuli used in emotion research, like image, music, metal imagery, and films.

The interest of exposing subjects to a various type of stimulus is to evoke emotions, which can be achieved by watching an emotional movie, listing to pieces of music or watching expressional photos.

Various international photos/sounds databases are used in the experimental investigation of emotions. Most popular ones are IAPS (International Affective Picture System) and IADS (International Affective Digital Sounds) which provide normative ratings of emotion (pleasure, arousal, dominance) for a set of acoustic stimuli.

In work [1], 2 experiments were conducted to evoke emotions, the first using labeled audio stimuli from the IADS database, where 27 clips were chosen to induce five emotional states: 3 clips for neutral, 6 clips for positive/low aroused, 6 clips for positive/high, 6 clips for negative/low and 6 clips for negative/high; in the second experiment, music stimulus was achieved by playing music pieces: a piece per emotion.

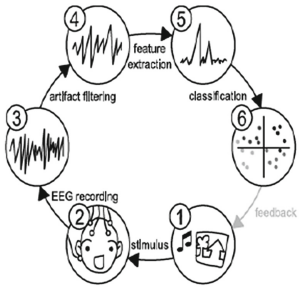


Fig. 2. Emotion recognition process [12]

In order to label the recorded EEG with the corresponding emotions, participants were asked to fill a questionnaire.

On other test experiments, the procedure was realized by a paradigm using stimuli from IAPS. Subjects were presented with 54 pictures (18 per state) [7].

Furthermore, in work [10], participants were stimulated by viewing images selected from IAPS. The selection of three image subsets, corresponding to the emotional states of interest, was based on the imperial thresholds on the valence and arousal scores. This selection was composed of these set of pictures: 106 for calm state, 71 for positive exciting state and 150 for negative exciting state.

3.2 EEG Recording

EEG recording is achieved by neuro headsets. Some of them are aimed to R&D purposes and are used in clinical diagnosis and trials. Others are made available for commercial use, such as the EMOTIV headset, which is the most widely used in recent researches.

In many studies [1,2,8–10], the 14-channels, wireless EMOTIV is used, with 128 Hz as sampling rate and 16 bits as resolution. The subject was exposed to 5 trials where each composed of happy and unhappy stimulus. Subjects are recommended to not move in order to eliminate artifacts related to movements.

Some of them proposed the Self-Assessment Manikin (SAM) to evaluate the real filling of volunteers.

Other researches drew more advantages from neuro-caps having more EEG channels, similar to work [5] where EEG data were collected from 250 gel-based scalp electrodes, plus four infraocular and two electrocardiographic (ECG) placements using a BioSemi ActiveTwo amplifier with 24-bit resolution. Caps with a custom whole head montage that covered most of the skull, forehead, and lateral face surface were used.

In work [3] the recording of EEG signals has been done through Nervus EEG, with 63 channel electrodes at a sampling frequency of 256 Hz. Meanwhile, EEG recordings in study [4] were conducted using the g.MOBIlab (g.tec medical & electrical engineering) portable biosignal acquisition system.

3.3 Preprocessing

In biomedical signal processing, the determination of noise and artifacts in the acquired signal is necessary in order to minimize their influence while extracting features.

The origin of an artifact can be physiologic, generated by other source than the brain such as ocular (electroculogram: EOG) which are most dominant below 4 Hz, muscle (electromyogram: EMG) above 30 Hz and heart rate (electrocardiogram: EMG) around 1.2 Hz. They can also be extraphysiologic, meaning, they arise from outside the body and are in the high 50 Hz [12], it can be related to the EEG acquisition settings or the environment [13].

Many methods such as band-pass filtering and artifact cancelling have been applied in several researches to determine these artifacts and extract interesting EEG waves. In work [4], EEG signals were firstly extracted during picture projection, then a band-pass Butterworth filter 8–30 Hz was applied to the resulted signals in order to extract waves of interest, which are, in this work as well in [10], alpha (8–12 Hz) and beta (13–30 Hz), finally the filtered data was segmented into 5 s segments, each correspond to the duration of picture projection.

Whereas, study [8] used Blind Source Separation (BSS) provided by Independent Component Analysis (ICA) to filter ocular and muscular artifacts, then a comparison has been realized to deduce that EEG raw data after artifacts removal gives better accuracy than data from direct recording.

3.4 Features Extraction and Selection

Based on features vector building during this step, classification, which is the next step in emotion recognition process, is made possible.

In many researches, linear transformation to EEG signals was applied to extract features, others followed nonlinear transformation, based on the fact that EEG signal is a chaotic and nonlinear signal.

Features are generally EEG signals characteristics. The most widely known is Power Spectral Density (PSD). In addition to Common Spatial Pattern (CSP),

Higher Order Crossings (HOC), Spectral Power Asymmetry (ASM), Higher Order Spectra (HOS), Asymmetric Spatial Pattern (ASP), Fractal Dimension (FD) and statistical features, have all shown good result in several researches.

In work [2], PSD was chosen for its little computation time which is suitable to real-time emotion recognition implementation. PSD value was computed from each EEG extracted band, to be used as feature. Study [11] realizes a comparison of performance for three features extraction techniques which are statistical features, Power Spectral Density (PSD) and High Order Crossing (HOC). Using KNN to evaluate the performances, PSD and HOC achieved better accuracy than Statistical features: 70.1%, 69.6% and 66.25% respectively. These results are consistent with others obtained based on the same features extraction techniques: PSD [15], HOC [16,17], statistical features [18,19].

In works [1,20], FD value was calculated from one electrode FC6 to recognize the arousal level used as feature, and EEG signals collected from electrodes AF3 and A4 situated in left/right hemisphere were used to calculate FD values for valence level recognition by applying Higuchi fractal dimension algorithm to the filtered data.

3.5 Classification

As previously mentioned, one of the recognition system stages have been carried out, and assuming that the extracted features are suitable to distinguish different emotion states, and to decide the class to which features belong to, a classification stage is thus required.

The objective of Classification is to describe a boundary between classes and to label them based on their measured features. Classifying can be done simply by fixing a threshold for each feature, or sophisticated, by using machine learning algorithms.

Many machine learning algorithms have been used as emotion classifiers such as Support Vector Machine (SVM), Nave Bayes (NB), Quadratic Discriminant Analysis (QDA), K-Nearest Neighbors (KNN) and Linear Discriminant Analysis (LDA) as shown in Table 2.

SVM is widely used in recent EEG-based emotion recognition researches, for it being one of the most popular supervised learning algorithms for solving classification problems. Many approaches are developed to face multi-class classification with SVM which was originally designed for binary classification problems, such as in study [6] which achieved a 92.57% in recognizing four emotions: Joy, Angry, Sadness and Pleasure. While in work [2], SVM was used for binary classification, in order to recognize two emotions: happy and unhappy. The average accuracy achieved was 70.55%. Also as a binary classification, SVM was investigated in work [20], to recognize high and low arousal levels by the arousal feature as well as positive and negative levels by valence features. The results were 100% as the higher performance and the worst was around 70%. This accuracy is a bit better than obtained by [21,22].

The k-Nearest neighbor (KNN) has shown high accuracy in classifying emotions. In work [7], two KNN with k=1,3 were trained to classify emotions into

negative, neutral and positive. The result was better with 1NN than 3NN: 72.22% and 62.96% respectively. *Bastos-Filho et al.* in their study has implement the KNN classifier with k=1,..,8 to evaluate and compare three features extraction. This study demonstrated that better results are always obtained with 8 neighbors and depends to the used features vector. The best accuracy achieved was 70.1% using PSD feature [11].

4 Discussion

As shown in Table 2, stimulus, techniques of features extraction, classifiers and its parameters are different depending on the main objective of each study.

The rates of emotion recognition depends on many factors: first of all, the number of emotions to classify, as shown, best results have been achieved by researches where emotions are not various [2,6,8,10] while in [1,3] where 6 emotions are recognized, the accuracy was not indicated. The choice of the couple (feature, classifier) influences the obtained results (i.e. HOC with SVM gives better accuracy than HOC with QDA [4]).

Compared to other stimuli, emotional films present several advantages. The existing studies have already evaluated the reliability and efficiency of film clips to elicitation [28]. Going over this, Virtual Reality (VR) technology have the ability to simulate complex and real situations. One important application of VR is the investigation of pathological processes in mental disorders, especially anxiety disorders, which is the main objective of our project.

Several studies using EMOTIV Epoc neuro-headset have shown its efficiency in the EEG recording stage [2,9,10]. EMOTIV with 14 EEG channels plus 2 references based on the international 10–20 electrode placement system operates at a resolution of 14 bits per channel with frequency response between 0.16–43 Hz [29].

It is proved that EEG signals belong to nonlinear and chaotic signals [30,31], many studies are based on this justified assumption in choosing the technique of features extraction. *Lui et al.* in their research [1] capitalize on this signal characteristic to justify the choice of Fractal Dimension (FD) as feature. In this study, there was no chosen classifier, FD values were compared to a predefined threshold in order to recognize the emotion, although the result is not mentioned, the use of one of the famous classifiers can be beneficent. In addition to study [14], a 70.5% as rate of recognition of five emotions was achieved by combining FD and SVM.

As shown in this table, the Real-Time implementation was not proposed by many researches, even though it represents an important constraint when implementing Emotion Recognition in games, diagnosis or stress and anxiety applications.

EEG-based Emotion Recognition is mostly used in Emotional Avatars and Games. While this system can be beneficial if implemented in mental diagnosis and decisional systems, the exploitation of Emotion Recognition should expand more in medical and clinical areas.

Table 2. Summary of features extraction techniques and classifiers

Reference Year	Participant	Neuro-headset	Emotions	Stimulus	Features	Classifier	Results	Real-Time
[3] 2008	6	Nervus EEG	6: Happy Disgust Fear Sad Surprise anger	Audio-visual Movie Clip	Statistical Features: Energy MEE RMS	FCM	-	NO
[6] 2009	26 subjects	32-channels EEG module	4: Angry Joy Pleasure Sadness	Music	PSD and ASM12	SVM	92.57%	NO
[7] 2010	20 subjects	NeXus-32 amplifier	Positive Neutral Negative	Pictures from IAPS	LPP	KNN	72.22%	NO
[1] 2011	22 Students	EMOTIV	6 Sad Frustrated Fear Satisfied Pleasant Happy	Audio stimulus with IADS Music stimulus	FD	Threshold	-	YES
[9] 2012	1 volunteer	EMOTIV Epoc	5: Happy Angry Sad Relaxed Neutral	Pictures from IAPS	HFD	SVM	70.5%	YES
[10] 2012	5 subjects	EMOTIV Epoc	3: Positively excited Neutral Negatively excited	Pictures from IAPS	HOC	KNN	90.77%	NO
[14] 2012	20 subjects	-	5: Happy Angry Sad Relaxed Neutral	15 Pictures	FD	SVM	70.5%	YES
[2] 2013	10 subjects	EMOTIV	2: Happy Unhappy	100 pictures from GAPED and 2 classical music pieces	PSD	Gaussian SVM	75.62%	YES
[8] 2013	11 participants	EMOTIV	2: positive negative	100 pictures from GAPED	PSD	SVM	85.41%	NO
[24] 2015	12 volunteers	30 channels EEG with neuroscan4.5 amplifier system	5: Neural Happy Sad Tense Sisgust	15 Movie Clips: 3 per emotion	PSD and AI	SVM	93.31%(PSD) 85.39%(AI)	NO
[25] 2015	4 subjects	B-alert X10From ABM	2: Pleasant and Unpleasant	40 pictures: 20 per emotion	Features based on ERDS/ERD	SVM KNN	59.9%(SVM) 57.35%(KNN)	NO
[28] 2015	15 subjects	62-channel electrode cap with ESI NeuroScan System	3: Positive Neutral Negative	15 emotional video clips	Differential entropy(DE)	DBN SVM LR KNN	86.08%(DBN) 83.99%(SVM) 82.70%(LR) 72.60%(KNN)	NO
[26] 2015	Preprocessed Dataset	-	4: Happy Relax Sad Fear	-	PSD	SVM	88.51%	NO
[27] 2016	DEAP Datasets	32-Channels BCI device	-	Musical video records from DEAP	Statistical features, PSD and FD	SVM	60.7%(Arousal), 62.33%(Valence)	NO

Feature: Power Spectral Density **(PSD)**, Spectral Power Asymmetry **(ASM)**, Higher Order Crossings **(HOC)**,Asymmetry Index **(AI)**,event-related synchronization/desynchronization **(ERDS/ ERD)**, Fractal Dimension **(FD)** and late positive potential **(LPP)**.
Statistical Features: Recoursing Energy Efficiency **(REE)**, Root Mean Square **(RMS)**.
Classifier: Support Vector Machine **(SVM)**, Nave Bayes **(NB)**, Quadratic Discriminant Analysis **(QDA)**, K-Nearest Neighbors **(KNN)**, Linear Discriminant Analysis **(LDA)**, Multilayer Perceptron **(MLP)**, Filter bank common spatial pattern **(FBCSP)**, Deep belief Networks **(DBNs)**, and Artificial Neural Network **(ANN)**.
Others: Advance Brain Monitoring **(ABM)**.

5 Future Work

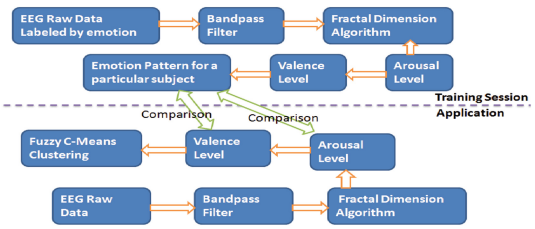


Fig. 3. Emotion recognition procedure

Based on the literature review of recent researches for Emotion Recognition techniques, and considering characteristics of EEG signals presented previously, we propose in this section, a combination of techniques. By implementing this architecture we aim to achieve better rates in Emotion Recognition. We will implement one

of the Fractal Dimension Algorithms in our system to test its efficiency when combined with Fuzzy C-Means classifier.

The Fuzzy C-Means technique is proved to be more general and useful in case of overlapping clusters. It is based on minimization of an objective function. The generalization is acceptable only when large sets of samples are available for classification. We will capitalize on these characteristics to prove that FCM combined with FD can achieve better accuracy in recognizing emotions in real-time than recent studies. As shown in Fig. 3, the EEG raw data collected from the EMOTIV Epoc headset will be preprocessed in order to remove noise and artifacts, in this stage a band-pass filter will be applied to extract the bands of interest. Then, a fractal dimension algorithm will be applied to the filtered data for the calculation of FD values that will represent the Features Vector (FV). The latter will be exploited to distinguish the arousal level, when the valence level of emotions is recognized within the similar arousal level. Finally, and based on these results (arousal and valence levels), emotions will be mapped into a 2D model and Fuzzy C-means clustering will be used for classification. During the training session, the same steps will be applied to an EEG raw data labeled by emotions to have an emotion pattern as result.

6 Conclusion

Approaches in the field of Emotion Recognition differ within the main study objective. Emotion stimulation methods, EEG-recording systems, features extraction techniques and classifiers are all dependent, therefore best choices are essential to achieve high performance in Emotion Recognition.

In order to respect many requirements such as Real-Time recognition, we propose, as future work, to use Fractal Dimension values (FD) as features and Fuzzy C-Means as a classifier in order to achieve high performances in recognizing emotions. An architecture was proposed and the choice of techniques that will be investigated in each Emotion Recognition step are justified.

Furthermore, the use of Virtual Reality (VR) can show better performance in emotion elicitation than traditional methods called in similar researches.

Acknowledgment. The authors would like to acknowledge the financial support of this work by grants from General Direction of Scientific Research (DGRST), Tunisia, under the ARUB program.

References

1. Liu, Y., Sourina, O., Nguyen, M.K.: Real-time EEG-based Emotion Recognition and its Applications (2011)
2. Jatupaiboon, N., Pan-ngum, S., Israsena, P.: Real-Time EEG-based happiness detection system. Sci. World J. (2013)
3. Murugappan, M., Sazali, Y., Hazry, D.: Lifting schema for humanemotion recognition using EEG. In: International Symposium on Information Technology (2008)

4. Petrantonakis, P.C., Hadjileontiadis, L.J.: Emotion recognition from EEG using higher order crossings. IEEE Trans. Inf. Technol. Biomed. **14**(2), 186–197 (2010)
5. Kothe, C., Onton, J., Makeig, S.: Emotion recognition from EEG during self-paced emotional imagery. In: Affective Computing and Intelligent Interaction (2013)
6. Lin, Y.P., Wang, C., Wu, T.L., Jeng, S.K., Chen, J.H.: EEG-based emotion recognition in music listening: a comparison of schemes for multiclass support vector machine. In: IEEE ICASSP (2009)
7. Schuster, T., Gruss, S., Kessler, H., Scheck, A., Hoffmann, H., Traue, H.: EEG: pattern classification during emotional picture processing. In: 3rd International Conference on PErvasive Technologies Related to Assistive Environments, vol. 67 (2010)
8. Jatupaiboon, N., Pan-ngum, S., Israsena, P.: Emotion classification using minimal EEG channels and frequency bands. In: The 10th International Joint Conference on Computer Science and Software Engineering, May 2013
9. Anh, V.H., Van, M.N., Ha, B.B., Quyet, T.H.: A real-time model based support vector machine for emotion recognition through EEG. In: ICCAIS (2012)
10. Haiyan, X., Konstantinos, N.: Affect recognition using EEG signal. In: MMSP Conference (2012)
11. Bastos-Filho, T.F., Ferreira, A., Atencio, A.C., Arjunan, S., Kumar, D.: Evaluation of feature extraction techniques in emotional state recognition. In: The 4th International Conference on Intelligent Human Computer Interaction (IHCI) (2012)
12. Bos, D.O.: EEG-based Emotion Recognition: The Influence of Visual and Auditory Stimuli, pp. 1–17 (2006)
13. Benbadis, S.R., Rielo, D.A., Talavera, F., Alvarez, N.: EEG Artifacts (2015)
14. Anh, V.H., Van, M.N., Ha, B.B., Quyet, T.H.: A real-time model based support vector machine for emotion recognition through EEG. In: ICCAIS 2012 (2012)
15. Lin, Y., Wang, C.-H., Jung, T., Wu, T.: EEG-based emotion recognition in music listening. IEEE Trans. Biomed. Eng. **57**(7), 1798–1806 (2010)
16. Petrantonakis, P.C., Hadjileontiadis, L.J.: EEG-based emotion recognition using hybrid filtering and higher order crossings. In: 3rd International Conference on Affective Computing and Intelligent Interaction and Workshops (2009)
17. Petrantonakis, P., Hadjileontiadis, L.: A novel emotion elicitation index using frontal brain asymmetry for enhanced EEG-based emotion recognition. IEEE Trans. Inf. Technol. Biomed. **15**(5), 737–746 (2011)
18. Schaaff, K., Schultz, T.: Towards an EEG-based emotion recognizer for humanoid robots. In: The 18th IEEE International Symposium on Robot and Human Interactive Communication, pp. 792–796 (2009)
19. Mampusti, E.T., Ng, J.S., Quinto, J.J.I., Teng, G.L., Suarez, M.T.C., Trogo, R.S.: Measuring academic affective states of students via brainwave signals. In: 3rd International Conference on Knowledge and Systems Engineering (2011)
20. Sourina, O., Liu, Y.: A fractal-based algorithm of Emotion Recognition from EEG using Arousal-Valence Model. In: Proceedings of the International Conference on Bio-inspired Systems and Signal Processing, January 2011
21. Zhang, Q., Lee, M.: Analysis of positive and negative emotions in natural scene using brain activity and GIST. Neurocomputing **72**, 1302–1306 (2009)
22. Kulish, V., Sourin, A., Sourina, O.: Analysis and visualization of human electroencephalograms seen as fractal time series. J. Mech. Med. Biol. World Sci. **26**(2), 175–188 (2006)
23. Lang, P.J., Bradley, M.M., Cuthbert, B.N.: International Affective Picture System (IAPS): technical manual and affective ratings. The Center for Research in Psychophysiology. University of Florida, USA (2005)

24. Lui, S., Meng, J., Zhang, D., Yang, J., Zhao, X., He, F., Qi, H., Ming, D.: Emotion recognition based on EEG changes in movie viewing. In: 7th Annual International IEEE EMBS Conference on Neural Engineering (2015)
25. Hou, C.L., Mountstephens, J., Teo, J.: EEG-based recognition of positive and negative emotions using for pleasant vs. unpleasant images. Int. J. Recent Adv. Multi. Res. **02**(06), 181–485 (2015)
26. Lokannavar, S., Lahane, P., Gangurde, A., Chidre, P.: Emotion recognition using EEG signals. Int. J. Adv. Res. Comput. Commun. Eng. **5**(5), 54–56 (2015)
27. Atkinson, J., Campos, D.: Improving BCI-based emotion recognition by combining EEG feature selection and kernel classifiers. Int. J. Expert Syst. Appl. **47**(C), 35–41 (2016)
28. Zheng, W.-L., Lu, B.-L.: Investigation critical frequency bands and channels for EEG-based emotion recognition with deep neural networks. IEEE Trans. Auton. Ment. Dev. **7**(3), 162–175 (2015)
29. Emotiv EPOC/EPOC+. https://emotiv.com/epoc.php
30. Kulish, V., Sourin, A., Sourina, O.: Human electroencephalograms seen as fractal time series: mathematical analysis and visualization. Comput. Biol. Med. **36**, 291–302 (2006)
31. Russell, J.A.: A circumplex model of affect. J. PSP **39**(6), 1161–1178 (1980)
32. Collura, T.F.: The Measurement, Interpretation, and Use of EEG Frequency Bands (1997)

Text Segmentation with Topic Modeling and Entity Coherence

Adebayo Kolawole John(✉), Luigi Di Caro, and Guido Boella

Dipartimento di Informatica, Universita Di Torino,
Corso Svizzera 185, 10149 Torino, Italy
kolawolejohn.adebayo@unibo.it, {dicaro,guido.boella}@di.unito.it

Abstract. This paper describes a system which uses entity and topic coherence for improved Text Segmentation (TS) accuracy. First, Linear Dirichlet Allocation (LDA) algorithm was used to obtain topics for sentences in the document. We then performed entity mapping across a window in order to discover the transition of entities within sentences. We used the information obtained to support our LDA-based boundary detection for proper boundary adjustment. We report the significance of the entity coherence approach as well as the superiority of our algorithm over existing works.

Keywords: Text segmentation · Entity coherence · Linear dirichlet allocation · Topic modeling

1 Introduction

The goal of Text Segmentation (TS) is to identify boundaries of topic shift in a document. Discourse structure studies have shown that a document is usually a mixture of topics and sub-topics. A shift in topics could be noticed with changes in patterns of vocabulary usage [14]. The process of dividing text into portions of different topical themes is called Text Segmentation [16]. The text units (*sentences* or *paragraphs*) making up a segment have to be coherent, i.e., exhibiting strong grammatical, lexical and semantic cohesion [18]. Applications of TS includes Information Retrieval (IR), passage retrieval and document summarization [1].

Our approach is an unsupervised method which also incorporates the use of topics obtained from LDA topic modeling of some documents. Furthermore, we incorporate entity coherence [2], that allows the introduction of some heuristic rules for boundary decision. The remaining parts of the paper describes the proposed system. In Sect. 2, we describe the general text segmentation task and related works. Section 3 details the proposed system followed by evaluation and results on choi's TS dataset.

A. Abraham et al. (eds.), *Proceedings of the 16th International Conference on Hybrid Intelligent Systems (HIS 2016)*, Advances in Intelligent Systems and Computing 552,
DOI 10.1007/978-3-319-52941-7_18

2 Background and Related Works

A document is a mixture of topics spread across its constituent words, sentences and paragraphs. The dimension of shift in topics is thus a function of the semantic bond and relationships within these units. Observingly, this bond tends to be higher among units with common topics. This notion is what is termed *cohesion* or *coherence* within a document. Cohesion is a function of grammatical factors, e.g., co-reference and sentential connectives as well as lexical factors like collocation [18]. Coherence is higher within units that share several topics. The goal of TS is to identify points of weak or no coherence in a text.

Text Segmentation could be Linear or Hierarchical. Unlike hierarchical Text Segmentation [12] which is more fine-grained, Linear TS algorithms [3,8,16] observes sequence of topic shifts without considering the sub-topic structures within segments. Past works have relied on the use of similarity in vocabulary usage in sentences in order to detect potential topic shift [8,16]. This idea, otherwise known as *lexical cohesion* could be tricky as it suffers from *lexical ambiguity*. This is because there are usually more than one words available to express an idea, i.e., *synonyms* while some words have multiple meanings, i.e., *polysemy*. The use of topics has recently been proposed [9–11,28], inspired by distributional semantics based approaches such as Latent Sementic Analysis (LSA) [9,19] and LDA topic models [22,28]. Previous works on Text Segmentation basically adopt two approaches, e.g., lexical cohesion and discourse based techniques [9]. In the former, lexical relationships that exist between contiguous text units are used as a measure of coherence. These lexical relationships include vocabulary overlap which could be identified by word stem repetition, context vectors, entity repetition, word frequency model and word similarity [3,15,18,26,29]. High vocabulary intersection between two compared units is taken to mean high coherence and vice versa. The TextTiling algorithm [16] excels in this category. It assigns a score to each topic boundary candidate within k chosen window. Topic boundaries are placed at the locations of valleys in this measure, and are then adjusted to coincide with known paragraph boundaries. The authors in [9] builds on this ideas with the introduction of a similarity matrix neighborhood ranking, where the rank of an element corresponds to the number of neighours with lower values.

The discourse-based techniques rely on the use of cue phrases and Prosodic features, e.g., pause duration that are most probable to occur close to a segment boundary. These features are combined using a machine learning model [3,24,26]. This approach however is domain independent and can only perform well if the system is evaluated on documents which uses the same cue words.

Recent works [11,22,28] employed topic modeling with LDA [4]. The idea is to induce the semantic relationship between words and to use frequency of topic assigned to words by LDA instead of the word itself to build sentence vector. This makes sense since a word could appear under different topics thus partially overcoming lexical ambiguity.

Similarly to these works, our implementation uses topics obtained with the LDA topic model. However, we introduced two heuristics (*lexical* and *semantic*) strictly for boundary adjustment. For instance, a position *m+1* after a sentence

S_m is a valid boundary only if sentences within the region S_{m-k} and S_{m+k} have no common entities, where k is chosen window. Also, coherent sentences tend to have similar semantics. This is the main idea in TextTiling and Choi's work [8,15] with the exception that they rely on term frequency to build sentence vector used for similarity calculation. Since this approach suffers from lexical ambiguity, e.g. the word *dog* appearing in one sentence followed by *puppy* in another are not deemed to be similar, we incorporate a semantic-net based similarity using WordNet. This typically overcomes the *synonymy* problem for a more efficient similarity calculation. The two heuristics were combined in a way to help in boundary decision making with topics-based sentence similarity. The experiment conducted on Choi's text segmentation evaluation dataset has shown the competitiveness of our approach.

3 Approach Description

Given an input document **W**, our algorithm divides the document into a set of minimal text units (s_1, s_2, s_3, ..., s_T), where T is the number of sentences in the document, each s_i can be viewed as a pseudo-document that contains a list of tokens v ∈ V, where V is the set of vocabulary of *W*. In practice, the goal is to identify sets of contiguous s_i that are mono-thematic, each member of the set being a segment.

Following similar works [11,22], we employ LDA topic modeling algorithm [4,5] to obtain topics for each word. Topic models are a suite of unsupervised algorithm that uncovers the hidden thematic structures in document collection. Modeling documents based on topics provides a simple way to analyze large volumes of unlabelled text while exposing the hidden semantic relationships between them.

3.1 LDA Basics

LDA is a generative probabilistic model of a corpus with the intuition that a document is a random distribution over latent topics, where each topic is characterized by a distribution over words in the vocabulary. Say for instance that a document is perceived as a bag of words where the order does not matter, suppose that the fixed number of topics (say for instance n_T) is known. Considering there could be many of such documents in a bag, then each word in the bag is randomly assigned a topic t drawn from the Dirichlet distribution. This gives a topic representations of the documents and word distributions of all the topics. The goal is then to find the proportion of the words in document **W** that are currently assigned to each topic t as well as the proportion of assignments to topic t over all documents that come from this word w. In other words, a Dirichlet distribution of each word over each topic is obtained. The model has shown capability to capture semantic information from documents in a way similar to probabilistic latent semantic analysis [17] such that a low dimensionality

representation of texts is produced in the semantic space while preserving their latent statistical features.

More formally, Given a document $\mathbf{w}$ of N words such that $\mathbf{w}$ = (w_1,w_2, w_3...w_N) and a corpus D of M documents denoted by D = ($\mathbf{w}_1$,$\mathbf{w}_2$,$\mathbf{w}_3$.....$\mathbf{w}_M$). For each of the words w_n in the document, a topic z_n is drawn from the topic distribution θ, and a word w_n is randomly chosen from $\mathrm{P}(\mathrm{w}_n \mid \mathrm{z}_n, \beta)$ conditioned on z_n. Given α, a k-vector with components with $\alpha_i > 0$ and the Gamma function $\Gamma(\mathrm{x})$. The probability density of the Dirichlet is given as

$$P(\Theta|\alpha) = \frac{\Gamma(\sum_{i=1}^{k} \alpha_i)}{\Pi_{i=1}^{k} \Gamma(\alpha_i)} \Theta_1^{\alpha_1 - 1} .... \Theta_k^{\alpha_k - 1} \tag{1}$$

Given the parameters α and β, the joint distribution of a topic mixture θ, a set of N topics $\mathbf{z}$, and a set of N words $\mathbf{w}$ is thus given by

$$P(\theta, \mathbf{z}, \mathbf{w}|\alpha, \beta) = P(\theta|\alpha) \Pi_{n=1}^{N} P(z_n|\theta) P(w_n|z_n, \beta) \tag{2}$$

Integrating over θ and summing of z, the set of topic assignments, the distribution of a document can be obtained as below

$$P(\mathbf{w}|\alpha, \beta) = \int P(\theta|\alpha) \left(\Pi_{n=1}^{N} \sum_{z_n} P(z_n|\theta) P(w_n|z_n, \beta) \right) \mathrm{d}\theta \tag{3}$$

where $\mathrm{P}(\mathrm{z}_n \mid \theta)$ is θ_i for the unique i such that $z_n^i = 1$ The probability of a corpus is obtained through the product of marginal probability above for each $\mathbf{w}_n$ in D as given below:

$$P(\mathbf{w}|\alpha, \beta) = \left\{ \Pi_{d=1}^{M} \int P(\theta_d|\alpha) \left(\Pi_{n=1}^{N_d} \sum_{z_d n} P(z_d n|\theta_d) P(w_d n|z_d n, \beta) \right) \mathrm{d}\theta_d \right\} \tag{4}$$

Training the LDA model on a corpus requires feeding the model with sets of tokens from the document. The model statistically estimate the topic distribution θ_d for each document as well as the word distribution in each topic. A model can also be used to predict topic classes for a previously unseen document. We trained the LDA algorithm with a mixture of the a subset of the wikipedia data, Brown corpus and Choi's dataset [8].

3.2 Topics-Based Sentence Similarity

The authors in [27] used the most frequent topics assigned to a word after the gibbs inference to avoid instability associated with a generative algorithm like the LDA. Contrarily, for each sentence, we obtain the distribution of topics for each word[1], together with their probability score and simply choose the topic with highest probability for each word. For each sentence, this results into a bag

[1] Our system is being developed in the context of our bigger project Eunomos [6,7].

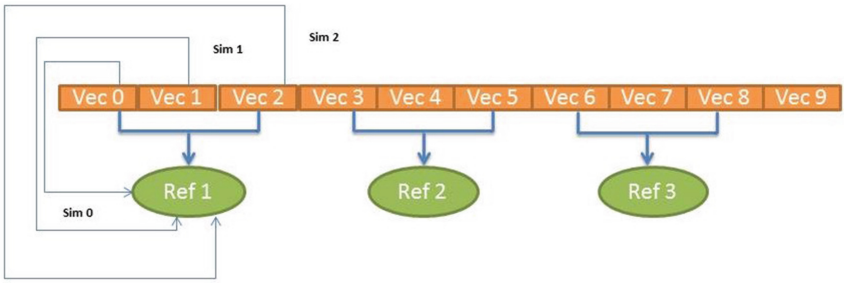


Fig. 1. Summing over window vector

of topics where order does not matter. We obtain a matrix $G = L \times T$ where $l \in L$ is a vector of length k, the chosen number of topics. Each vector l contains the frequency of each topic ID assigned by the LDA to the words in a sentence, where by topic ID, we denote the topic group or cluster that a word belongs, i.e., a number in the range $[0,\ T-1]$. As an example, assuming the number of topics n = 10 and the bag of topics for a sentence is $\{0, 0, 5, 2, 3, 3, 7, 7, 1, 6, 5\}$, then the vector for such a sentence will be [2,1,1,2,0,2,1,2,0,0], each element representing the frequency of occurrence of topics 0 to 9. A generally accepted assumption is that sentences with similar topics have some semantic relationship. Furthermore, the LDA is able to unravel the latent relationship between words through its probabilistic clustering.

We introduce a lookahead window w_n which has a value of 3 by default. This is similar to the *k-block* of sentences employed in [28] but with different objective. The previous works compares the vector of a sentence to the *k-block* of sentences on the left and the right of the sentence in order to get the similarity score[2] for that sentence. The process is then carried out for all sentences in the document in order to yield the measure of closeness of a sentence to its surrounding sentences. In our implementation, for each pass over the list of sentences, using the lookahead window, we sum up the vectors of sentences within the window and use it as a *reference* vector for sentences within that window. The intuition is that we can treat the set of sentences within a window as a *mini* document, summing up the vectors gives the overall meaning of the *mini* document. Thus, we can estimate the semantic distance between the *mini* document and each neighour sentence. Sentences with high topic correlation will have high similarity to the reference. Figure 1 shows the process of summing over vector for a sample document of 10 sentences. Once the reference values have been obtained, the next step is to obtain sentence similarity, otherwise called the coherence score. To do this, for each window, we use the cosine similarity between each sentence and the reference vectors. Repeating this over all sentences results into a time series, e.g., a one dimensional vector of similarity values over all the sentences.

[2] Otherwise called coherence score.

3.3 Inter-Sentence Semantic Similarity

To further alleviate the language variability problem, we introduce another similarity vector. First, we perform parts of speech (POS) tagging[3] on the sentences in order to select the verbs, nouns and the adjectives. We call this the *POS profile* of each sentence. Here, we also rely on the use of the lookahead window. For instance, using the WordNet concept hierarchy, we calculate the similarity of the POS profile of a sentence with available sentences within a shifting window of 3. As an example, given the verbs, nouns and adjectives in a sentence S_1, instead of comparing these POS entries directly with those in sentence S_2 only, it is compared with those sentences that falls into the set $\{S_2, S_3, S_4\}$. To derive similarity from WordNet, we used both the path length between each word as well as the depth function. Our similarity implementation is similar to the approach in [20] and produces a score within the range 0 and 1 for each compared POS filtered sentences. Similarly, we obtain a 1-D similarity vector with length equal to the number of sentences.

3.4 Entity-Based Coherence

An observation well established in grounded theories of coherence [13,21] in discourse analysis is that entity distribution and transition signals coherence. The works in [2] is based on the centering theory, where the authors represents a document as a grid of entities in the document with their roles (subject, object, neither subject nor object and absence) specified as the actions of these entities. The rows of the grid correspond to sentences, while the columns correspond to discourse entities. We follow this ideas by observing the spread of entities across the sentences in the document to be segmented. Contrary to the grid-based entity ranking [2], our goal is to observe the entity overlaps that exist between sentences within a chosen shift window[4]. Succinctly, we only use the information about entity coherence for necessary boundary adjustment and not boundary detection to be specific. To achieve this, we use a grammar-based Regex parser to extract all the noun phrases in each sentence. To determine the overlap for a sentence S_i, we compute the ratio of its common noun-phrases to its right neighours within a specified window, e.g., $\{S_{i+1}, S_{i+2}, S_{i+3}\}$. The entity overlap is obtained as follows:

$$\mathrm{EOV} = \frac{|A \,\tilde{\cap}\, B^*|}{|A \cup B^*|} \tag{5}$$

where A and B^* represent the set of entities in the sentence being considered and right neighors within a specified window, respectively. The intersection, $\tilde{\cap}$, allows partial matches since the entities are considered equivalent if there is an exact match or an entity is a substring of the other. Instead of using the overlap score, we record the last sentence from within the B^* that has shared entities

[3] We used the Stanford POStagger. It is available at http://nlp.stanford.edu/software/tagger.shtml.

[4] Following our previous parameter w_n, we use a window of 3 sentences as default.

with A if the overlap score actually exceeds a threshold. As an example, if a sentence S_1 is compared to $\{S_2, S_3, S_4\}$ with the entity overlap score between them exceeding the threshold, then, one by one, we check if it actually has an overlap with each of S_2, S_3 and S_4 independently. If say for instance, we discover that S_1 and S_4 do not have any common entities but it has with S_2 and S_3, then the index of sentence S_3[5] is used as its last sentence collocation. It becomes plain whether a sentence share entities with immediate neighbors in which case the assumption is that such a sentence is not likely to be a boundary. As an example, the text below shows how entity coherence may support boundary adjustment. The entities detected by our custom parser are in bold.

S_1: ***Cook*** *had discovered a* ***beef*** *in his possession a few days earlier and, when he could not show the* ***hide****, arrested him.*
S_2: *Thinking the evidence insufficient to get a conviction, he later released him.*
S_3: *Even while suffering the trip to his* ***home****,* ***Cook*** *swore to* ***Moore*** *and* ***Lane*** *that he would kill the* ***Indian****.*
S_4: *Three weeks later, following his recovery, armed with a* ***writ*** *issued by the* ***Catskill justice*** *on* ***affidavits*** *prepared by the* ***district attorney****,* ***Cook*** *and* ***Russell*** *rode to arrest* ***Martinez****.*
S_5: *Arriving at daybreak, they found* ***Julio*** *in his* ***corral*** *and demanded that he surrender.*
S_6: *Instead, he whirled and ran to his* ***house*** *for a* ***gun****, forcing them to kill him,* ***Cook*** *reported.*

In the example above, the entity *Cook* appears in S_1, S_3, S_4 and S_6. Considering S_1, we conclude that no boundary exist until S_4 since there is significant entity overlap with S_3 and S_4 when moving over the sentence window. Even though there appears to be no overlap with S_2 and S_1, it is safe to assume that S_2 is not a boundary since it falls within a coherent window, same goes for S_5 which falls within sentences S_3 and S_6. In our implementation, we create a vector whose elements holds the index of the last sentence it has an overlap with. In case of no overlap, the entry for a sentence is set at 0. Identifying the entity distribution in this way is useful for boundary adjustment for the suggested boundary from our topic based segmentation.

3.5 Boundary Detection and Segmentation

To obtain the sets of possible segmentation from the coherence score vectors, we obtained the local minima (valleys) and the local maxima (peaks). The valleys are the smallest values within a local range of the coherence scores vector. Since coherence scores are higher within sentences sharing many topics, we assume that these points of minimum values signals the points where least topic cohesion

[5] We use index here to mean the unique ID of a sentence, e.g., sentence 1 will have index 0, sentence 2 will have index 1 etc..

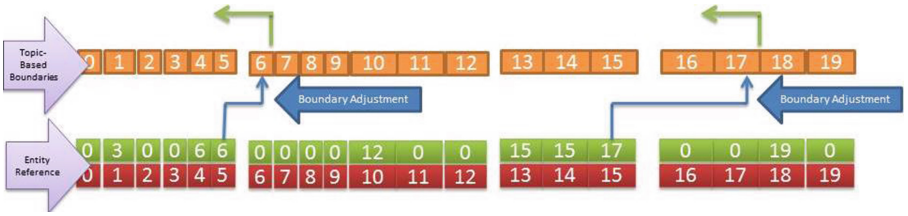


Fig. 2. Entity coherence-based boundary adjustment

occurs, hence a segment boundary. The indices of the valleys[6] are collected in a vector as potential points of topic shift. We use the entries from our entity based coherence for necessary boundary adjustment. A mapping between the topic-based vector and the entity-coherence vector is created. For each sentence in a document, each column of the entity coherence vector references the index of the last sentence it has an overlap with. If there is a boundary after a sentence but there is an overlap reference to a sentence index higher than the boundary point then we *left-shift* the boundary as an adjustment task. Figure 2 shows the process of boundary adjustment over a sample sentence. The idea is based on centering theory [2], sentences with overlapping entities above a threshold have some level of coherence.

4 Evaluations

For all evaluations, we used the Choi's dataset since it allows easy comparison with our baseline systems [8,16,28] In order to evaluate the accuracy of our system, we used the P_k error [3] and WindDiff [25] evaluation metrics which are commonly used. These two metrics measures the rate of error in segmentation with a lower value signifying better segmentation accuracy. Other common metrics are the IR based precision, recall and accuracy. However, these IR based metrics over-penalizes the *near miss* scenarios, e.g., when an actual segment is wrongfully partitioned into two different segments by an algorithm.

We trained the LDA model on the Brown corpus and a trimmed version of Wikipedia dump[7]. We used the Gensim version of the LDA algorithm. Gensim is a python library for an array of NLP tasks[8]. The number of Topics specified for training is 50 with 20 inference iterations.

We compared the result of our algorithm with the TopicTiling system [28], a TextTiling based system which solely rely on topics assignment to document from LDA. We also compared the result with TextTiling and Choi's system as

[6] i.e., the vector index which corresponds to the index of each sentence in the local minima.

[7] The wikipedia dump was downloaded on July 30, 2015. It is accessible at https://dumps.wikimedia.org/enwiki/.

[8] It is available at https://radimrehurek.com/gensim/.

Table 1. P_k error metrics on Choi's dataset

Window	3–5	6–8	9–11	3–11
1	1.76	2.90	4.0	2.64
3	0.89	1.18	0.49	0.67
5	1.30	1.53	3.80	1.80

Table 2. WinDiff error metrics on Choi's dataset

Window	3–5	6–8	9–11	3–11
1	1.82	2.94	4.21	2.68
3	0.93	1.41	0.49	0.71
5	1.29	1.48	3.87	1.82

Table 3. Comparison of our systems's performance with selected state of the arts algorithm

Algorithm	3–5	6–8	9–11	3–11
TextTiling	44	43	48	46
Choi LSA	12	9	9	12
Topic Tiling	1.24	0.76	0.56	0.95
Our System	0.89	1.18	0.49	0.67

Table 4. P_k error metrics on Choi's dataset without boundary adjustment

Window	3–5	6–8	9–11	3–11
1	1.92	3.30	4.1	2.98
3	1.19	2.23	0.82	0.91
5	1.70	2.36	3.89	2.20

reported by Rield and Bielmann [27]. For all the reported results from other systems, we did not reproduce the experiments, relying on the results reported in [27].

Tables 1 and 2 shows the results of our algorithm on Choi's Text Segmentation dataset using the P_k and WinDiff error metrics, respectively. Table 3 gives the comparison of our system against some state-of-the-art systems. Specifically, we selected TopicTiling [27] algorithm as it is the most similar to our work. Our intention is to show that our boundary-adjustment ideas really improves the performance of the system. The TextTiling and Choi's work have been severally outclassed by other systems [11,22,23] but were selected based on their popularity. The TopicTiling algorithm has also shown slight superiority over the latter algorithms. To show the importance of the boundary adjustment, we repeated the experiment without adjusting the boundary. Table 4 shows the effect of the boundary adjustment. Note the decrease in performance when boundary adjustment is not used.

5 Conclusion

We presented a TS approach that outperforms famous state-of-the-art systems on the Choi's TS dataset. Our approach combines the use of topics for segmentation with Entity Coherence-based heuristics for an improved performance. For the topic-based segmentation, we used the popular topic modeling algorithm, LDA. We described the approach of obtaining the coherence scores of the sentences. The reported results confirm the competitiveness of our approach.

Acknowledgments. Kolawole J. Adebayo has received funding from the Erasmus Mundus Joint International Doctoral (Ph.D.) programme in Law, Science and Technology. Luigi Di Caro and Guido Boella have received funding from the European Union's H2020 research and innovation programme under the grant agreement No 690974 for the project "MIREL: MIning and REasoning with Legal texts".

References

1. Barzilay, R., Elhadad, M.: Using lexical chains for text summarization. In: Advances in Automatic Text Summarization, pp. 111–121 (1999)
2. Barzilay, R., Lapata, M.: Modeling local coherence: an entity-based approach. Comput. Linguist. **34**(1), 1–34 (2008)
3. Beeferman, D., Berger, A., Lafferty, J.: Statistical models for text segmentation. Mach. Learn. **34**(1–3), 177–210 (1999)
4. Blei, D.M., Lafferty, J.D.: Dynamic topic models. In: Proceedings of the 23rd International Conference on Machine Learning, pp. 113–120. ACM (2006)
5. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent dirichlet allocation. J. Mach. Learn. Res. **3**, 993–1022 (2003)
6. Boella, G., Di Caro, L., Humphreys, L., Robaldo, L., Rossi, R., van der Torre, L.: Eunomos, a legal document and knowledge management system for the web to provide relevant, reliable and up-to-date information on the law. Artif. Intell. Law **24**(3), 245–283 (2016)
7. Boella, G., Di Caro, L., Ruggeri, A., Robaldo, L.: Learning from syntax generalizations for automatic semantic annotation. J. Intell. Inf. Syst. **43**(2), 231–246 (2014)
8. Choi, F.Y.Y.: Advances in domain independent linear text segmentation. In: Proceedings of the 1st North American Chapter of the Association for Computational Linguistics Conference, pp. 26–33. Association for Computational Linguistics (2000)
9. Choi, F.Y.Y., Wiemer-Hastings, P., Moore, J.: Latent semantic analysis for text segmentation. In: Proceedings of EMNLP. Citeseer (2001)
10. Dias, G., Alves, E., Lopes, J.G.P.: Topic segmentation algorithms for text summarization and passage retrieval: an exhaustive evaluation. In: AAAI, vol. 7, pp. 1334–1339 (2007)
11. Du, L., Pate, J.K., Johnson, M.: Topic segmentation in an ordering-based topic model (2015)
12. Eisenstein, J.: Hierarchical text segmentation from multi-scale lexical cohesion. In: Proceedings of Human Language Technologies: the 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics, pp. 353–361. Association for Computational Linguistics (2009)
13. Grosz, B.J., Weinstein, S., Joshi, A.K.: Centering: a framework for modeling the local coherence of discourse. Comput. Linguist. **21**(2), 203–225 (1995)
14. Alexander, M., Halliday, K., Hasan, R.: Cohesion in English. Routledge (2014)
15. Hearst, M.A.: Texttiling: a quantitative approach to discourse segmentation. Technical report. Citeseer (1993)
16. Hearst, M.A.: Texttiling: segmenting text into multi-paragraph subtopic passages. Comput. Linguist. **23**(1), 33–64 (1997)
17. Hofmann, T.: Probabilistic latent semantic indexing. In: Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 50–57. ACM (1999)

18. Kaufmann, S.: Cohesion and collocation: using context vectors in text segmentation. In: Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics on Computational Linguistics, pp. 591–595. Association for Computational Linguistics (1999)
19. Landauer, T.K., Foltz, P.W., Laham, D.: An introduction to latent semantic analysis. Discourse Process. **25**(2–3), 259–284 (1998)
20. Li, Y., McLean, D., Bandar, Z.A., O'shea, J.D., Crockett, K.: Sentence similarity based on semantic nets and corpus statistics. IEEE Trans. Knowl. Data Eng. **18**(8), 1138–1150 (2006)
21. Mann, W.C., Thompson, S.A.: Rhetorical structure theory: toward a functional theory of text organization. Text-Interdiscip. J. Study Discourse **8**(3), 243–281 (1988)
22. Misra, H., Yvon, F., Cappé, O., Jose, J.: Text segmentation: a topic modeling perspective. Inf. Process. Manage. **47**(4), 528–544 (2011)
23. Misra, H., Yvon, F., Jose, J.M., Cappe, O.: Text segmentation via topic modeling: an analytical study. In: Proceedings of the 18th ACM Conference on Information and Knowledge Management, pp. 1553–1556. ACM (2009)
24. Passonneau, R.J., Litman, D.J.: Discourse segmentation by human and automated means. Comput. Linguist. **23**(1), 103–139 (1997)
25. Pevzner, L., Hearst, M.A.: A critique and improvement of an evaluation metric for text segmentation. Computat. Linguist. **28**(1), 19–36 (2002)
26. Reynar, J.C.: Statistical models for topic segmentation. In: Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics on Computational Linguistics, pp. 357–364. Association for Computational Linguistics (1999)
27. Riedl, M., Biemann, C.: Text segmentation with topic models. J. Lang. Technol. Comput. Linguist. **27**(1), 47–69 (2012)
28. Riedl, M., Biemann, C.: Topictiling: a text segmentation algorithm based on LDA. In: Proceedings of ACL 2012 Student Research Workshop, pp. 37–42. Association for Computational Linguistics (2012)
29. Utiyama, M., Isahara, H.: A statistical model for domain-independent text segmentation. In: Proceedings of the 39th Annual Meeting on Association for Computational Linguistics, pp. 499–506. Association for Computational Linguistics (2001)

Deep Learning for Hot Topic Extraction from Social Streams

Amal Rekik[1,2(✉)] and Salma Jamoussi[1,2]

[1] Multimedia InfoRmation Systems and Advanced Computing Laboratory, MIRACL, Sfax University, Sfax, Tunisia
[2] Digital Research Center of Sfax, DRCS, Sfax, Tunisia
rekik.amal91@gmail.com, jamoussi@gmail.com

Abstract. Extracting hot topics from data streams is one of the most exciting tasks that interest the researchers in the social networks field. To achieve that goal, many studies focused on robust data stream clustering algorithms. In this paper, we propose an evolving method to extract hot topics from social networks called SAE-Clus. This work explores a deep learning technique that provides interesting advantages especially in the context of data streams. Our method attempts to meet the principal requirements of data stream clustering algorithms. To evaluate the performance of our proposed methods, experiments were conducted using the "Sanders" and "HCR" datasets.

Keywords: Social network · Topic extraction · Data streams · Clustering · Deep learning

1 Introduction

Social networks are focused on sharing information on a very large scale. In the context of news diffusion, extensive studies have been made. In fact, in social media, users can filter, access and react to information at any moment. Today, because of the large number of users and the huge amount of information entered as data streams, processing and analyzing such data is a challenge for researchers in data mining. Indeed, a data stream is defined as a sequence of non-stationary, potentially infinite data items $(x^{(1)}, x^{(2)}, x^{(3)} \ldots x^{(n)})$ where the objects of the stream $x^{(i)} \in R^n$ can be represented by a set of attributes with a vector of length d, $x^{(i)} = [x_1^{(i)}, x_2^{(i)}, x_3^{(i)} \ldots x_d^{(i)}]$. The data stream can be distinguished from static data by a set of characteristics like the high volume of continuous data, the rapid rate of arriving data, the changes over time of the data distribution and the presence of noisy data.

In order to extract hot topics from data streams, many researchers have been focusing on different clustering methods to classify data into topics and to highlight the major topics. In fact, clustering is the process of partitioning sets of unlabeled data into a series of homogenous subsets called clusters, maximizing the similarities among objects in the same cluster and minimizing the similarities among objects in different clusters. Due to the high speed, the huge size of data set and their dynamic evolving characteristics, the traditional data classification and the learning techniques that are designed to process fixed size batches of data are inefficient to analyze such evolving

A. Abraham et al. (eds.), *Proceedings of the 16th International Conference on Hybrid Intelligent Systems (HIS 2016)*, Advances in Intelligent Systems and Computing 552,
DOI 10.1007/978-3-319-52941-7_19

data with an unbounded size and a high speed rate. For this purpose, clustering algorithms for data streams raise fundamental issues to be addressed. The challenging requirements of the continuous data clustering algorithms are the ability to reflect the nature of the data and to adjust models to the new constantly arriving data. Such algorithms should also meet with the requirements of speed and memory.

Achieving data stream clustering in order to detect hot topics is the main focus of this paper. For this purpose, we propose a new evolving method of clustering called SAE-Clus, that explores deep learning techniques to learn incrementally from unlabeled examples generated at high speed and that need to be classified instantly.

The reminder of this paper is planned as follows: The next section is dedicated to the related work. In Sect. 3, we describe the exploitation of Stacked Autoencoders with data streams. In Sect. 4, we go into details of our approach. Section 5 reports the experiments results and discussions. Finally, Sect. 6 concludes the paper.

2 Related Work

Data streams entered from social networks are continuously evolving over time and arriving at a high speed rate. For this purpose, mining such data requires learning techniques that are efficient to analyze evolving data having a huge volume and an unknown feature space. In fact, several researchers have recently analyzed the text data shared on social media and pursued the same goal as ours of extracting hot topics. This section discusses these previous works and highlights the most relevant algorithms and methods.

2.1 Data Stream Clustering Algorithms

Chen and Tu [1] proposed an algorithm called DenStream for discovering clusters of arbitrary shape in an evolving data stream where the core-micro-cluster is introduced to summarize the clusters while the potential core-micro-cluster and outlier micro-cluster structures are proposed to maintain and distinguish the potential clusters and outliers. However, DenStream requires adjusting many parameters to have a good clustering result. Authors, in [2], proposed a clustering algorithm called Dstream which uses a grid-based approach. It maps each input data into a grid and computes the grid density. Then, all grids are clustered based on their densities using a density-based algorithm. D-stream adopts a density decaying technique to capture the changes over the data stream but it is not appropriate for the text data. In [3], Marcel et al. proposed the StreamKM++ clustering algorithm for data streams based on k-Means++ algorithm. So, they use a standard streaming technique called merge-and-reduce to maintain their samples in data streams. In this research, data is organized in samples and every time when two samples representing the same number of input points exist, authors take the union (merge step) and create a new sample (reduce step). However, this algorithm takes long computing time. In [4], the proposed CluStream algorithm is separated into online and offline components. It continuously maintains a fixed number of micro-clusters. The online component periodically stores detailed summary statics and the offline component uses only this summary statics and performs clustering using the

k-means algorithm. However, CluStream predefines a constant number of micro-clusters and it is also risky when the data contains noise. In [5], an ensemble learning approach for data stream clustering method have been proposed called Stream Ensemble Fuzzy C-Means SEFCM. This approach is composed of three stages. First, to decrease the number of samples in every step of processing, the data stream is divided to smaller blocks. Then, the ensemble clustering algorithm EFCM is executed on each block to cluster it and produce a partition. Finally, the concluding partitions are combined by considering the clusters centers produced by the previous step as new input data set and applying the EFCM algorithm on them, and as such, the produced partition is considered as the absolute one. The problem with this algorithm is the overall time complexity.

2.2 Hot Topics Extraction Methods

Various methods have been proposed dedicated to the problem of extraction of hot topics on data stream. For example, Tongyu et al. proposed, in [6], a pre recognition model for hot topic discovery using the concepts of the topic life cycle, the hot velocity and the hot acceleration. This model aims at discovering emerging hot topics before they boost and break out. It follows three steps. First, the original posts are clustered using the Latent Dirichlet Allocation LDA [7] and the Pachinko Allocation Model PAM [8], to get topics and their amounts. Then, the velocity and acceleration of the topic are calculated to find the different periods of the topic life cycle. Finally, potential hot topics during the boost period are selected through checking their amount, velocity and acceleration. In [9], an online version of LDA was proposed. This version consists of automatically capturing thematic patterns and identifying topics of text streams and their changes over time. In [10], Abid et al. proposed a novelty detection method in data stream clustering called AIS-Clus that uses the artificial immune system-meta heuristic. It consist of, for each incoming data, a scoring function is calculated in order to examine the affinity value of each cluster. The Artificial Immune System is involved to determine which cluster will absorb this novel invader by applying both the clonal selection principal which is based on the production of antibodies to best match the new invader, and the negative selection mechanism that enables detecting noisy data. However, AIS-Clus results are not constantly good since it is based on a set of arbitrary parameters.

Despite the theoretical advantages of the methods mentioned above, most of them did not meet all of the requirements of data stream clustering algorithms: some researches did not handle concept drift, some had lack in terms of accuracy and others were not concerned with the problem of the vocabulary evolution. Thus, they cannot be very suitable in the context of data streams and social networks.

3 Stacked Autoencoders

An Autoencoder (AE) neural network is an unsupervised learning algorithm that applies back propagation on a set of unlabeled training examples by setting the target value to be equal to the inputs. It is a two-layer neural network with one hidden layer.

The first layer is the encoding layer and the second is the decoding layer [11]. The main goal of the AE is to learn a representation for a set of data, typically, for dimension reduction [12]. It is trained to encode the input x into a representation c(x), so that the input can be reconstructed from that representation. Starting with random weights, the autoencoder can be trained by minimizing the discrepancy between the original data and its reconstruction. It takes an input vector $x \in [0,1]^d$, and maps it to a hidden representation $y \in [0,1]^{d'}$. The encoding of this input vector is done by a linear mapping and a nonlinear activation function of the network:

$$y = \text{sigm}(Wx + b_1) = (1 + \exp(-(Wx + b_1)))^{-1}. \quad (1)$$

This function is parameterized by W which is a d'*d weight matrix and b_1 which is the encoding bias vector. The resulting representation y is then mapped back to a reconstructed vector $z \in [0,1]^d$. The encoding step is performed using a separate decoding matrix:

$$z = \text{sigm}(W'x + b_2) = (1 + \exp(-(W'x + b_2)))^{-1}. \quad (2)$$

This function is parameterized by W' which is the decoding matrix and b_2 which is the decoding bias vector. Each training $x^{(i)}$ is thus mapped to a corresponding $y^{(i)}$. The autoencoder is fine-tuned using back-propagation of error-derivatives to make its output as similar as possible to its input, minimizing the reconstruction error which is:

$$L(y, z) = \frac{1}{2}\sum_{i=1}^{m} ||z_i - y_i||^2. \quad (3)$$

Autoencoders can be stacked to form so called Stacked Auto-Encoder (SAE). This deep nature provides high level feature learning. In the SAE, each layer is trained separately with an Autoencoder. Figure 1 illustrate this iterative training procedure.

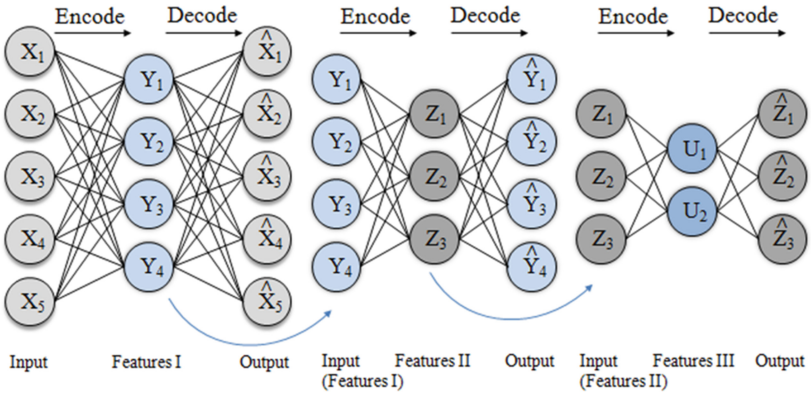


Fig. 1. Example of a stacked autoencoder

The first Autoencoder takes the input x_t and produces the representation $C^{(1)}(x_t)$ Then the second Autoencoder takes $C^{(1)}(x_t)$ as input and produces $C^{(2)}(x_t)$, and so on until $C^{(L)}(x_t)$, with L is the number of Autoencoders. Finally, all layers must be combined to form the stacked autoencoder with L-1 hidden layers.

Realizing that both AE and SAE are similar in terms of learning a representation for a set of data as well as for dimension reduction, SAEs have showed highest performance on several problems [13–15]. The advantage of this architecture is the use of more hidden layers than a single autoencoder which allows the high-dimensional input data to be reduced to a much smaller code representing the important feature.

4 Proposed Method

In this section, we present a detailed description of our proposed method called SAE-Clus to extract hot topics and how deep learning techniques are exploited for clustering text data streams. Data Streams consist of several updates and must be processed to extract the useful information instantly. Because of this high speed and this huge size of data set that arrives continuously, our proposed clustering algorithm should be efficient to manage the large amount of data, adaptive to manage the unknown change in the distribution of data and the vocabulary evolution and handles concept drift in real time. Our proposed algorithm consists to apply the stacked autoencoder to an evolving vector of features with unknown space, in order to reduce the dimensionality of data and improve the efficiency of the social streams clustering. SAE-Clus is composed of two phases:

The Static Phase. Historical data are clustered during this phase. Therefore, a step of preprocessing is achieved. Later, a representing and training step is realized. Then, the clustering algorithm k-means is used to obtain the initial clusters.

The Streaming Phase. It takes into account features selected and clusters obtained during the static phase. In this streaming phase, for each new arriving batch, the text is preprocessed, the features are evolved and the clusters are updated. Some clusters disappear, others grow and new clusters appear (Fig. 2).

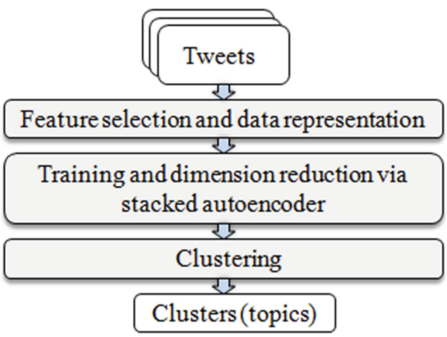


Fig. 2. Overall process of the proposed method

4.1 Feature Selection

The feature selection is one of the most interesting steps of our algorithm. It aims at identifying an optimal subset of interesting attributes to represent tweets after the text preprocessing step.

Static Phase. In this phase, a feature selection method is applied to select the best features. This is done by counting the number of tweets where a word appears (the frequency):

$$\text{Frequency}(\text{word}) = \frac{\text{NumberOfTweetsContainingWord}}{\text{TotalNumberOfTweets}}. \tag{4}$$

If the frequency of the word is greater than the user-defined threshold α, this word is considered as a feature. Then, we obtain the initial feature vector.

Streaming Phase. In the streaming phase, new words appear, so new features should be added with the progress of the stream. Thus, a feature evolution method is applied. Some attributes will be deleted and others will be added. For this purpose, for each batch, the frequencies of all words are calculated. Similar as in the static phase, if the frequency of the word is greater than the user-defined-threshold β, this word is considered as a feature. Indeed, the number of the frequent words that can be considered as features are limited by the user. If a previous feature is missed in the k previous batches (k is defined by the user), it is considered as no more relevant, and it must be removed from the feature vector. In fact, even if this word will return to be a feature during the following streams, it is considered as a new feature that concerns a new topic. The new features take the places of those removed in the feature vectors. If there is no place, new features will be added at the end of the feature vector.

4.2 Data Representation and Dimension Reduction

Static Phase. The SAE takes, for all the batches, an input vector having the same dimension. As the number of relevant words increases with the progress of data stream, this input vector dimension should be determined from the beginning. So, after the selection of all features, all tweets of this phase are represented by a binary vector that is formed depending on the presence of each attribute, previously selected, in each tweet. Thus, it has the same dimension as the feature vector. Then, the binary vector of each tweet is followed by a set of cases filled by zeros to form the SAE input vector with the determined dimension. Hence, a binary vector of the static phase is formed. It is considered as the representative vector of this phase. Regarding the high dimension of the data, the SAE takes as input all the tweets vectors of this phase, tries to learn them and returns the low-dimensional representations of the input vectors. When the SAE ends up learning, we give it as input the representative vector of this phase, and it returns its prediction as a low dimensional representation vector of the static phase, which will be used on the tweets vectors of the following batch.

Streaming Phase. In this phase, all tweets will be represented by a vector that is composed of two parts. The first part designs the low dimensional representation, produced by the SAE, of the representative vector of the previous stream. This part is meant to represent all previous data. The second part designs a vector of the representation of each tweet. This vector is formed depending on the presence of each attribute in the tweet. These two parts are followed by a set of cases filled by zeros to form the input vector of the SAE with the same dimension as in the static phase. Also, the representative vector of each stream is prepared at this step. At each batch, the SAE learns the tweets vectors and produces the low-representations of the tweets and the compressed representation of the representative vector of the current batch.

4.3 Clustering

Initial Clustering. Our algorithm consists of creating clusters from the low-dimensional representations obtained by the training of the SAE. For this purpose, the representations of the data in the static phase, produced by the SAE, are clustered using the k-means clustering algorithm. To solve the problem of the choice of the cluster number, we have proposed a method that calculates the intra-similarity between clusters. Then, it compares this intra-similarity with the user defined parameter S and decides the best number of clusters that should be produced in this phase. Thus, we obtain the initial clusters of tweets.

Evolving Clustering. For each batch, we achieve the clustering of the low-representations of the tweets obtained by the SAE with the same method of the static phase. Concerning the main goal of our method, we concentrate only on hot topics. Thus, with progress of stream, clusters (topics) will evolve. Some clusters will disappear, others will grow and new clusters will appear. So, a mapping step between clustering at time t and t−1 is required. First, a fusion parameter is calculated between each cluster C1, produced by the clustering at time t, of the current batch and every cluster C2, produced by the clustering at time t−1, of the previous batch. In fact, in each batch, every tweet is represented by the cluster, to which it belongs and the words composing the tweets belonging to this cluster. Later, every cluster is represented by a set of words which, the tweets within this cluster contains. So, the fusion parameters can be calculated as follow:

$$\mathrm{FP}(\mathrm{C1},\mathrm{C2}) = \frac{\mathrm{NumberComonWords}(\mathrm{C1},\mathrm{C2})}{\mathrm{TotalNumberWords}(\mathrm{C1})}. \tag{5}$$

$$\mathrm{FP}(\mathrm{C2},\mathrm{C1}) = \frac{\mathrm{NumberComonWords}(\mathrm{C2},\mathrm{C1})}{\mathrm{TotalNumberWords}(\mathrm{C2})}. \tag{6}$$

The counting of the fusion parameter is, for one raison, to conclude how much two topics from two following streams are similar as they share a set of the same words, and consequently, if they should be merged. If one of those fusion parameters between two

clusters is greater than the user-defined-fusion threshold γ, these clusters are considered as similar. So, they are specific to the same topic. Thus, they will be merged in the same cluster. Also, cluster that did not merged during the λ previous streams will be no more considered as a relevant one since it is representing a topic which is no more discussed. Moreover, the number of posts talking about a new topic, different to the existing ones, can increase in a short time. Therefore, a new cluster representing this new topic will appear. The final clusters obtained are the clusters that represent the hot topics in social networks. Figure 3 details the process steps of the streaming phase.

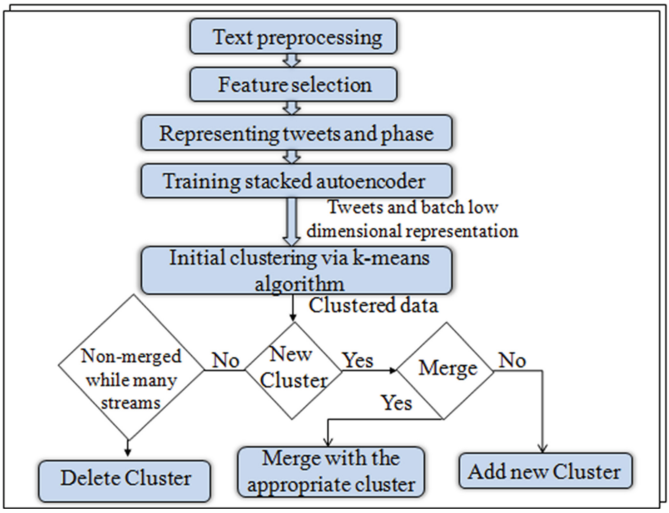


Fig. 3. The overall streaming clustering process

5 Experiments

In order to evaluate the performance of our algorithm, we conducted several experiments based on the two datasets which have been widely used in text stream clustering algorithms and will be firstly introduced: Sanders [16] and Health Care Reform HCR [16]. We compare our results with three of the most relevant algorithms DenStream [1], CluStream [3] and Dstream [2].

5.1 Training Datasets

The training datasets which are used in this evaluation are divided into two phases, static and streaming. The dataset Sanders consists of 5,512 hand-classified tweets about four different topics: Apple, Google, Microsoft and Twitter [16]. After the preprocessing step to filter the English tweets, 3065 tweets are obtained.

The second dataset HCR is an annotated dataset based on tweets about health care reform in the USA that contains the hashtag «#hcr» [17]. It consists of 2516 tweets

Table 1. Sanders data set

Phases	Samples	Topics	Samples per topic
Static	365	Apple	180
		Google	185
Streaming	2700	Apple	686
		Google	603
		Microsoft	797
		Twitter	619

Table 2. HCR data set

Phases	Samples	Topics	Samples per topic
Static	330	Hcr	180
		Obama	150
Streaming	450	Democrats	130
		Gop	120
		Conservatives	80
		Liberals	30
		Teaparty	20
		Hcr	70

with 8 different targets: Hcr, Obama, Democrats, Gop, Conservatives, Liberals, Teaparty and Stupak (Tables 1 and 2).

5.2 Evaluation of the Proposed Method

In order to evaluate the performance of our proposed approach, we carry out a set of experiments. In all experiments, we use the stacked autoencoder with seven hidden layer = 7, and for each autoencoder 8000 iterations except with the static phase where we fix 9000 iterations. First, we perform the SAE with our data sets. We start by providing it with the set of tweets vectors of the static phase and looking at the low-dimensional representation of the obtained results that will be evaluated later. Therefore, we use the SVM model to classify these low-dimensional representations and estimate its capacity of learning and discovering interesting low dimensional representation of the input vectors. The results are as follows: (Table 3).

Table 3. SVM results

	Sanders	HCR
SVM with binary representations	49%	60%
SVM with SAE representations	99%	91%

As detailed in the previous table, the SVM model is applied, first, to the feature vectors of the tweets during the static phase. The model has predicted correctly only

49% of the data for the Sanders data set and 60% for the HCR data set. Second, SVM model is applied to the low-dimensional representative vectors of the same tweets produced by the SAE. The model has predicted correctly 99% of the data for the Sanders data set and 91% for the HCR Data set. So, we can conclude that the stacked autoencoder is well trained since it produced interesting low-dimensional representation of the input vectors that will be clustered later.

5.3 Performance Comparison

For comparison purposes, we used the CluStream, DenStream and Dstream algorithms. We have fixed the size of the batch to 200 tweets for the Sanders dataset and 100 tweets for the HCR dataset. During the evolving phase, the accuracies vary each time where a new cluster is detected. These evolutions are depicted in Fig. 4 and the final results are tabulated in Table 4.

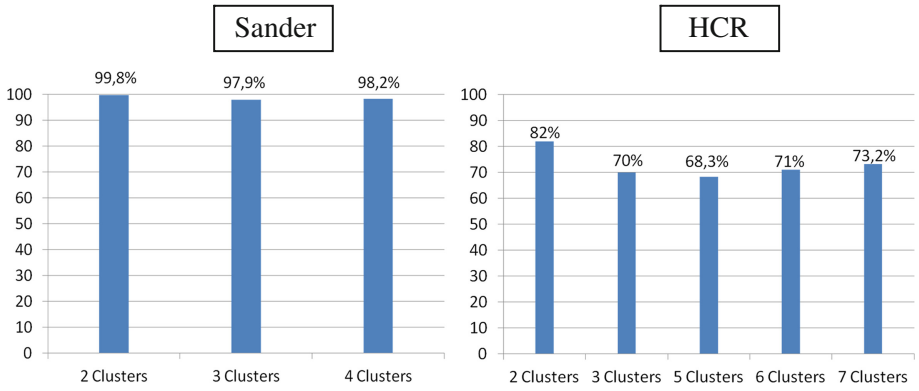


Fig. 4. The evolution of the accuracy with each new detected cluster

Table 4. Performance comparison

Datasets		CluStream	DenStream	Dstream	SAE-Clus
Sanders	Recall	0.95	0.63	0.62	**0.87**
	Precision	0.49	0.5	0.61	**0.89**
	F-measure	0.65	0.56	0.61	**0.88**
	Clusters number	4	17	442	**4**
HCR	Recall	0.88	0.98	0.65	**0.69**
	Precision	0.35	0.33	0.33	**0.75**
	F-measure	0.5	0.5	0.44	**0.72**
	Clusters number	7	1	26	**7**

Results show that performance has improved in terms of precision, F-measure and clusters number as compared with other algorithms in case of both Sanders and HCR Datasets and, thus, our method can extract hot topics from social streams efficiently.

6 Conclusion

In this article we presented a new method based on deep learning architecture to extract the most relevant topics from evolving social streams in an online manner. The basic approach consists of three procedures. First, data are represented in a scalable manner. Then, a stacked autoencoder is trained to produce low dimensional representation for the data which will be clustered later in a scalable way to detect the real time evolution of topics. The experiments prompt that our method can extract efficiently the hot topics and outperform two relevant data stream algorithms. We plan in the future to ameliorate the accuracy of our method with imbalanced datasets.

References

1. Chen, Y., Tu, L.: Density-based clustering for real-time stream data. In: Proceeding KDD 2007, pp. 133–142 (2007)
2. Cao, F., Ester, M., Qian, W., Zho, A.: Density-based clustering over an evolving data stream with noise. In: SDMM, pp. 328–339 (2006)
3. Ackermann, M.R., Martens, M., Raupach, C., Swierkot, K., Lammersen, C., Sohler, C.: StreamKM++: a clustering algorithm for data streams. ACM J. Exp. Algorithmics **17**(1) (2012)
4. Aggarwal, C.C., Watson, T.J., Ctr, R., Han, J., Yu, P.S.: A framework for clustering evolving data streams. In: VLDP, pp. 81–92 (2003)
5. Fathzadeh, R., Mokhtari, V.: An ensemble learning approach for data stream clustering. In: 21st Iranian Conference on Electrical Engineering (ICEE) (2013)
6. Zhu, T., Yu, J.: A prerecognition model for hot topic discovery based on microblogging data. Sci. World J. (2014) Hindawi Publishing Corporation
7. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent Dirichlet allocation. J. Mach. Learn. Res. **3**, 993–1022 (2003)
8. Li, W., McCallum, A.: Pachinko allocation: DAG-structured mixture models of topic correlations. In: Proceedings of the 23rd International Conference on Machine Learning (ICML 2006), Pittsburgh, USA, pp. 577–584, June 2006
9. AlSumait, L., Barbara, D., Domeniconi, C.: On-Line LDA: adaptive topic models for mining text streams with applications to topic detection and tracking. In: 2008 Eighth IEEE International Conference on Data Mining (2008)
10. Abid, A., Jamoussi, S.: Novelty detection in data stream clustering using the artificial immune system. In: European Mediterranean & Middle Eastern Conference on Information Systems 2016 (EMCIS 2016), Krakow, Poland (2016)
11. Lauzon, F.Q.: An introduction to deep learning. In: 11th International Conference on Information Science, Signal Processing and their Applications (ISSPA) (2012)
12. Bengio, Y.: Learning deep architectures for AI. Found. Trends Mach. Learn. **2**, 1–172 (2009)

13. Kandaswany, C., Silva, L.M., Alexandre, L.A., Santos, J.M.: High-content analysis of breast cancer using single-cell deep transfer learning. In: IEEE International Conference on Acoustics, Speech and Signal Processing (2013)
14. Gehring, J., Miao, Y., Metze, F., Waibel, A.: Extracting deep bottleneck features using stacked auto-encoders
15. Tan, C.C., Eswaran, C.: Performance comparison of three types of autoencoder neural networks. In: Second Asia International Conference on Modelling Simulation (AMS)
16. Saif, H., Fernandez, M., He, Y., Alani, H.: Evaluation datasets for twitter sentiment analysis a survey and a new dataset, the STS-Gold. In: 1st International Workshop on Emotion and Sentiment in Social and Expressive Media, Turin, Italy (2013)
17. Speriosu, M., Sudan, N., Upadhyay, S., Baldridge, J.: Twitter polarity classification with label propagation over lexical links and the follower graph. In: Proceedings of the EMNLP First Workshop on Unsupervised Lea, Edinburgh, Scotland (2011)

Multi-shot Human Re-identification via Gabor-LBP Based Video Covariance Descriptor

Bassem Hadjkacem(✉), Walid Ayedi, and Mohamed Abid

CES Research Unit, National Engineers School of Sfax,
Sfax University, Sfax, Tunisia
bassem.hadjkacem.tn@ieee.org, ayediwalid@yahoo.fr,
mohamed.abid@rnu.enis.tn

Abstract. Multi-shot human re-identification is a major challenge because of the large variations in a human's appearance caused by different types of noise such as occlusion, viewpoint and illumination variations. In this paper, we presented a novel Gabor-LBP based video covariance descriptor, called GL-VC descriptor, which considers image sequences to extract appearance features, captures moving regions of interest and find the correlation between video frames. Therefore, it implicitly encodes the described human motion by the integration of temporal information and decreases the effect of occlusion. To deal with the changes of view points and illumination, the Local binary pattern (LBP) operators and Gabor bank were integrated into the spatio-temporal covariance features. We evaluated our GL-VC approach on the publicly available CAVIAR4REID multi-shot dataset and demonstrated superior performance in comparison with the current state-of-the-art.

Keywords: Multi-shot human re-identification · Spatio-temporel features · Covariance matrix · Gabor · LBP · CAVIAR4REID dataset

1 Introduction

Undoubtedly, automated human re-identification (re-id) systems play a key role in several security video surveillance applications. Re-identifying the same person across a non-overlapping camera networks is particularly challenging, since inter-camera, illumination and appearance variations and occlusion are often very clear. Most current research works have addressed this problem by matching spatial appearance features (e.g., color and gradient histogram) in the single-shot setting (i.e., assuming only one image per person per camera view) [1, 2, 4, 6]. These features are intrinsically limited due to the effects of noise.

In several real world video surveillance applications, we have a set of images for each tracked person. So, for the multi-shot case, there are several images, which form an image set, for each identity in query and corpus domains [7]. As the pipeline of a re-id system may include human feature modeling and matching, the performance of most descriptors often relies on the extracted features. Some works were based on spatio-temporal (S-T) information or biometric features. We note that a great deal of research in motion

A. Abraham et al. (eds.), *Proceedings of the 16th International Conference on Hybrid Intelligent Systems (HIS 2016)*, Advances in Intelligent Systems and Computing 552,
DOI 10.1007/978-3-319-52941-7_20

recognition and human behavior has exploited extensively the S-T information of video frames [8, 12, 21]. Some motions, like gait, are specific biological features and can be modeled by signatures which differ from one human to another [3, 9, 11].

In this work, we proposed a framework that generates specific image sequences to apply the S-T approach to the multi-shot re-id. It extracts Gabor-LBP based video covariance (GL-VC) vectors, which provide modeling capabilities of human motions, by the integration of the temporal axis in the appearance features. The main contribution is to integrate the Gabor filter and the Local binary Pattern (LBP) operator with other appearance features and find the correlation between video frames in order to reduce the effects of noise as illumination and view point changes.

The rest of the paper was organized as follows. Section 2, described the related work on multi-shot human re-id. GL-VC descriptor approach was presented in Sect. 3. Finally, Sect. 4 detailed and discussed the obtained experimental results.

2 Related Works

Different multi-shot models for human re-id have been proposed in literature. In fact, Gheissari et al. [7] proposed a space-time graph that uses different images to group the similar S-T regions. A S-T segmentation is applied to reject contours that are considered unstable information over time. Then, a triangulated model person is used to manage the correspondence between the body parts and adjust the image model person. The appearance-based method, proposed by Bazzani et al. [13], condenses a set of frames of any person into a Histogram Plus Epitome (HPE) descriptor. It embeds global chromatic content via a histogram representation and local descriptions via an epitomic analysis. Then, the authors apply HPE at the asymmetry-based segmentation introduced by Farenzena et al. [6], giving rise to the Asymmetry-based Histogram Plus Epitome (AHPE) descriptor. Furthermore, the model proposed by Badagkar-Gala and Shah [18] is a part-based S-T appearance model which combines facial features and colors.

On the other hand, the covariance descriptor, introduced by [1], manages to combine several characteristics such as color, texture and shape and allows locating these features. A covariance matrix can be computed from any type of images. Some noises, such as rotation and illumination changes, are absorbed by the covariance matrix [2]. Therefore, it was applied to object detection by [1], object tracking, and face recognition by [20]. More recently, it was used by [17] in action recognition. Several research works like [2] have also applied the covariance descriptor in mono-shot human re-id, but have not integrated the S-T information for a multi-shot re-id. Based on the-state-of-the-art, we notice that exploring S-T information from image sequences is appreciated. Therefore, that is why a GL-VC descriptor was proposed to solve the problem of multi-shot human re-id by integrating the temporal axis.

The differences between our approach and others are: (1) We exploit all multi-shots of each person extracted from video streaming, without selecting, and coding human motion with appearance features to reduce the occlusion problem; (2) We integrate LBP operators and Gabor bank in video covariance descriptor, not in image descriptor, to reduce the effects of view point and illumination changes; (3) It produces directly a

fixed compact representation by fusing S-T features and finding the correlation between different features without a learning metric.

3 GL-VC Descriptor

Generally, when monitoring an area covered by a network camera, the regions of interest of each human, extracted from video streams, have not the same dimensions. Our framework starts the re-id process by pre-treatment step of the new image sequences (e.g. applied the histogram equalization, resized and concatenated the new frames …). Then, the correlations between successive images are exploited over time, which can give modeling capabilities of pedestrian motion. Therefore, the GL-VC descriptor integrates the 'time' parameter with other features creating S-T Covariance (S-T Cov) signatures [3]. To reduce the noise effect, like view point change and variation of illumination, we also integrate also an LBP operator and a Gabor banc in GL-VC descriptor. An overview of our approach is illustrated in Fig. 1. In the first part, the S-T features extraction step was presented.

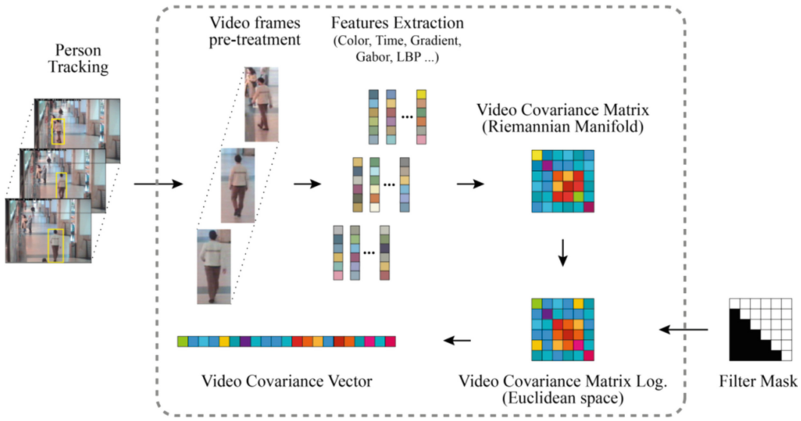


Fig. 1. Overview of the GL-VC descriptor for human re-identification

3.1 Spatio-Temporal Features Extraction

Let $p = (x, y, t)^T$ denote the horizontal, vertical, and temporal coordinates of a pixel. Let G denote the set of coordinates of all pixels belonging to an action segment (a group of pictures or a short video clip) which is X pixels wide, Y pixels tall, and T frames long. $G_{(X,Y,T)} := \{(x, y, t)^T : x \in [1, X], y \in [1, Y], t \in [1, T]\}$.

Let F be the X × Y × T × d dimensional feature group of pictures extracted from G:

$$F(x, y, t) = \emptyset(G, x, y, t) \tag{1}$$

Table 1. List of Gabor-LBP based video covariance features

	K	Features
Spatio-temporal	0	The **x** location of the corresponding region
Features	1	The **y** location of the corresponding region
LBP operators	**2**	**The time of the corresponding region**
Gabor banc	3	**I**, grayscale intensity value (the luminance component)
	4	**Cr**, color component value (the red chrominance component)
	5	**Cb**, color component value (the blue chrominance component)
	6	$\boldsymbol{I_x}(\boldsymbol{x},\boldsymbol{y},\boldsymbol{t})$, the norm of the first order derivatives in x
	7	$\boldsymbol{I_y}(\boldsymbol{x},\boldsymbol{y},\boldsymbol{t}) = \frac{\partial \boldsymbol{I}(\boldsymbol{x},\boldsymbol{y},\boldsymbol{t})}{\partial \boldsymbol{y}}$
	8	$\boldsymbol{I_y}(\boldsymbol{x},\boldsymbol{y},\boldsymbol{t})$, the norm of the first order derivatives in y
	9	$\boldsymbol{I_y}(\boldsymbol{x},\boldsymbol{y},\boldsymbol{t}) = \frac{\partial \boldsymbol{I}(\boldsymbol{x},\boldsymbol{y},\boldsymbol{t})}{\partial \boldsymbol{y}}$
	10	
	11	**Gradient**, $\theta(x,y,t) = \boldsymbol{tan}^{-1}\left(\frac{\boldsymbol{I_y}(\boldsymbol{x},\boldsymbol{y},\boldsymbol{t})}{\boldsymbol{I_x}(\boldsymbol{x},\boldsymbol{y},\boldsymbol{t})}\right)$
	12	
	13	**Magnitude**, $mag(x,y,t) = \sqrt{I_x^2(x,y,t) + I_y^2(x,y,t)}$
	14	**LBP,** R = 1, N = 8
	15	**LBP,** R = 2, N = 8
	16	**LBP,** R = 2, N = 16
	17	$\boldsymbol{Gabor_{00}}$, $O = 0$, $E = 0$
		$\boldsymbol{Gabor_{01}}$, $O = 0$, $E = 1$
		$\boldsymbol{Gabor_{02}}$, $O = 0$, $E = 2$
		$\boldsymbol{Gabor_{03}}$, $O = 0$, $E = 3$
		$\boldsymbol{Gabor_{04}}$, $O = 0$, $E = 4$

where the function $\emptyset$ can be a mapping time parameter with other features and k can be any feature such as color, time, etc. d represents the number of the used features and dimensions. We can arrange these features into two groups: structure features and content features. (Table 1) Let P be the X × Y × T × d tensor of the integral GOP.

$$P(X,Y,T,i) = \sum_{x \le X} \sum_{y \le Y} \sum_{t \le T} F(x,y,t)(i) \tag{2}$$

where F(x, y, t)(i) is the i^{th} element of vector F(x, y, t). Furthermore, let Q be the X × Y × T × d × d tensor of the second-order integral GOP:

$$Q(X,Y,T,i,j) = \sum_{x \le X} \sum_{y \le Y} \sum_{t \le T} F(x,y,t)(i).F(x,y,t)(j) \tag{3}$$

The covariance of the spatio-temporal GOP is;

$$CovG_{(X,Y,T)} = \frac{1}{S-1}[Q \quad - \quad \tfrac{1}{S}PP^T] \tag{4}$$

where;

$$\begin{cases} \mathrm{S} = \mathrm{x.y.t} \\ P = [P(x,y,t,1), \quad \cdots \quad P(x,y,t,d)]^T \\ Q = \begin{bmatrix} Q(x,y,t,1,1) & \cdots & Q(x,y,t,1,d) \\ \vdots & \ddots & \vdots \\ Q(x,y,t,d,1) & \cdots & Q(x,y,t,d,d) \end{bmatrix} \end{cases}$$

The S-T covariance region can be computed in $O(d^2)$ time. The dimension of the feature vector P is equal to d. So, P can be computed in d passes.

Since Q is a symmetric matrix of a d × d dimension, it can be computed in $(d^2 + d)/2$ passes. Therefore, for i, j = 1, …, d the computational complexity of constructing the integral GOP is $O(d^2XYT)$.

3.2 Gabor and LBP Features Extraction

To reduce the noise effects, we study the role of LBP and Gabor filters and features extraction methods.

(A) **Gabor Bank**

We can obtain Gabor features by convolving a 2-D Gabor wavelet with the image I(x, y) as follows:

$$G_{uv}(x,y) = |I(x,y) * \varphi_{u,v}(x,y)| \tag{5}$$

where |.| is a magnitude operator, u and v define the orientation and scale of the Gabor kernels.

The 2-D Gabor kernel is a product of an elliptical Gaussian and a complex plane wave (See Eq. 6) [20]. In a real scene of video surveillance, the distance between camera and people varies around a vertical axis. So, the vertical viewpoint change is smaller than horizontal one. To form the Gabor bank, the orientation of Gabor kernel 'u' equals 0. We take five different scales as Pang et al. [20] ($v \in \{0, \ldots, 4\}$). Thus, the Gabor bank is equal to $\{G_{00}, G_{01}, G_{02}, G_{03}, G_{04}\}$

$$\begin{cases} \varphi_{u,v}(z) = \frac{\|k_{u,v}\|^2}{\sigma^2} e^{\left(-\frac{\|k_{u,v}\|^2 \|z\|^2}{2\sigma^2}\right)} \left[e^{ik_{u,v}z} - e^{\left(-\frac{\sigma^2}{2}\right)}\right] \\ k_{u,v} = k_v e^{i\emptyset_u} \end{cases} \tag{6}$$

Where z = (x, y), ‖.‖ denotes the norm operator. $k_{u,v}$ denotes the wave vector $\emptyset_u = 0$, $k_v = k_{max}/f_v$. The value of the other parameters follows the setting used by Lui and Wechsler [14]: the maximum frequency $k_{max} = \frac{\pi}{2}$, the spacing factor between kernels in the frequency domain $f_v = \sqrt{2}$, the number of oscillations under the Gaussian envelope σ = 2π [16, 20].

(B) **LBP Operators**

The Local Binary Pattern (LBP) operator was introduced by Ojala et al. [15] as a texture descriptor. It encodes the relative intensity magnitude between each pixel and its neighboring pixels. It is invariant to the gray level change and can be efficiently extracted. LBP is widely used in several applications such as face recognition and person re-id [16]. In fact, Ying and Shutao [16] have used the LBP in person re-id in order to compensate the variation of illumination. The basic LBP operator is extracted and the decimal code of each pixel is directly assigned to the mapping function instead of forming a histogram.

We consider different neighborhood sizes of each pixel with two parameters; R means the region radius and N means the number of sampling points around the center. In our mapping function, we have chosen 3 cases of LBP operators; (First: R = 1, N = 8), (Second: R = 2, N = 8) and (Third: R = 2, N = 16). We integrated the Gabor bank and LBP operators with the S-T covariance features to reduce the impact of illumination and viewpoint changes and improve the recognition rate. Table 1 shows the set of features used in this work.

3.3 Distance Calculation Between Covariance Matrices

The covariance matrix is defined symmetric and positive. It can be represented as a connected Riemannian manifold but does not lie in the Euclidean space. The majority of the common machine learning methods work in Euclidean spaces. So, it is necessary to find a proper metric distance for measuring two covariance matrices. We used the 'Log-Euclidean Riemannian' Metric used in [3]. This metric defines a distance between two covariance matrices M1 and M2 as:

$$Dist(M_1, M_2) = ||\log\,(M_1) - \log\,(M_2)||_{\mathrm{F}} \tag{7}$$

where $\log(.)$ is the matrix logarithm, and $\|\cdot\|_{\mathrm{F}}$ is the Frobenius norm of a matrix.

We can map the covariance matrices from the Riemannian manifold on the Euclidean space using the matrix logarithm operation, given an (n x n) covariance matrix M. We apply the Singular Value Decomposition (SVD), which decomposes the matrix M into 3 matrices; $M = U\Sigma U^T$ where $\Sigma = diag(\lambda_1, \lambda_2, \ldots, \lambda_n)$ is the eigenvalues diagonal matrix, and U is an orthonormal matrix of size $d \times d$ By derivation, the covariance matrix $M^{'}$ in the Euclidean space is defined as:

$$\begin{aligned} M' &= \log(M) = \sum\nolimits_{i=1}^{\infty} \frac{(-1)^{i+1}}{i} (M - I_n)^k \\ &= U\,.\,diag(\log(\lambda_1), log(\lambda_2), \ldots, \log(\lambda_n))\,.U^T \end{aligned} \tag{8}$$

where I_n is an n × n identity matrix. As the covariance matrix is symmetric, we can apply a filter mask, extracting all the entries on and above or below the diagonal of the symmetric matrix, and representative it by a vector of $d \times (d+1)/2$ values [17]. So, GL-LBP descriptor transforms each image sequence of size $X \times Y \times T$ into a compact vector represented by 153 values.

4 Experimental Results

In this section, we presented an analysis the performance of GL-VC descriptor on «CAVIAR4REID» dataset. Extensive experiments on this dataset were conducted and a comparative study with several state of the art methods was presented.

4.1 Experimental Setup

As dataset, we used «CAVIAR4REID» to show the performance of GL-VC descriptor. «CAVIAR4REID» dataset contains 10 views for each of the 72 humans. It includes various types of noises such as occlusion, illumination and viewpoint variation, background changes … The original dataset of «CAVIAR4REID» consists of 26 real sequences recorded from two different points of view at the resolution of 384 × 288 pixels in a shopping centre. It includes people walking alone, meeting with others, window shopping, entering and exiting shops. The ground truth has been used to extract the bounding box of each human. It has broad changes in resolution. The minimum size of the images is 17 × 39 and their maximum size is 72 × 144 [19].

The performance of GL-VC descriptor is measured by the biometric identification method that returns ranked lists of candidates: Cumulative- Matching-Curve (CMC). Let G a gallery of m samples of different subjects, Q a probe set with n samples. We use all of them and determine the distance between two sets of descriptors extracted from G and Q by applying the nearest-neighbor classifier.

4.2 Evaluating the GL-VC Descriptor Parameters

In this section, we studied the impact of the Gabor and LBP features in GL-VC and the size of image sequences, on the recognition accuracy. First, we evaluated the recognition performances of a GL-VC descriptor with different combinations of Gabor and LBP operators on «CAVIAR4REID». We investigated the impact of integrating the Gabor and LBP features into the S-T covariance features on reducing the effect of illumination and view point changes. We normalized all video frames into 32 × 64 pixels and prepared the video sequences with selecting 5 frames per sequence for each human. Then, we applied the CMC metric on the image sequences with fixed sizes (e.g., Number of pictures per sequence: Np = 5) of all 72 pedestrian.

Figure 2 shows the recognition rates with different combinations. We found that the GL-VC based on full combination gives the best recognition rate in the first rank. In this case, the video covariance vector contains 145 values. So, we confirm that the image sequences of the Caviar dataset are very noisy with the changes of viewpoint and luminance and that several light sources intervened in the space of the cameras networks. On the other hand, the GL-VC descriptor can be applied on both small and large image sequences. Our approach treats any image sequence with various (Np) and provides fixed size vectors for the various sequence sizes. So, we studied the impact of the size of the image sequences, on the recognition accuracy. «CAVIAR4REID» is limited to 10 views per human; 5 views for Probe and 5 views for Gallery. We have chosen 3 cases for experimentation; Np is equal at 3, 4 and 5.

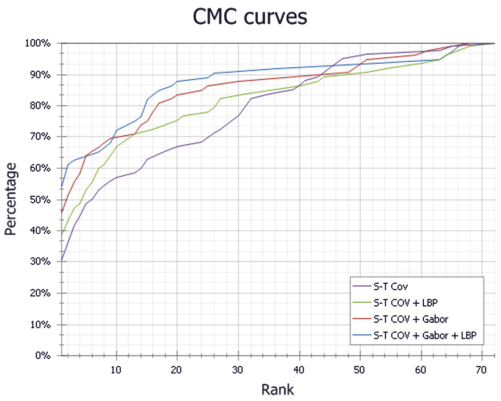


Fig. 2. CMC curves obtained by different combinations between S-T covariance descriptor and Gabor and LBP features

Figure 3 presents the performance of the GL-VC descriptor with various sizes of image sequences. Based on this figure, we notice that increasing the number of pictures in our descriptor allows reducing the effect of occlusion or change of background. Furthermore, in some rank values, the recognition rate of the smaller GOP is superior also to greater GOP. We believe that it exist added noises in the fourth or fifth frames. This result confirms that multi-shot human re-id using GL-VC descriptor can reduce the effect of some noises. But generally, we note that processing time with all features and the superior size is higher than the other cases. This choice increases the computational complexity of constructing the integral GOP which is O(d2XYT). (See part 3) So, we can make a compromise between the size of image sequences and the processing time according to a dataset or the camera network.

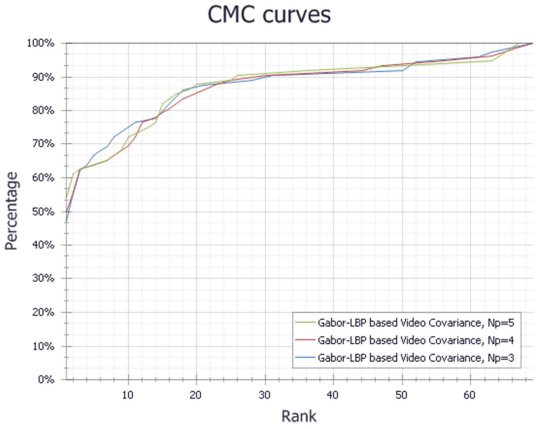


Fig. 3. CMC curves obtained by Gabor-LBP based video covariance descriptor on CAVIAR4REID for various sizes (Np = 3, 4, 5)

4.3 Comparison Experimental Results on the CAVIAR4REID Dataset with the-State-of-the-Art

First, when comparing our experimental results displayed in Figs. 2 and 3 and to the results of [5, 10, 13, 19], we can see that GL-VC method is more efficient in human recognition, when simulated on CAVIAR4REID dataset. Table 2 shows that the recognition rate in the first rank is equal to 53.4. So, the GL-VC approach gives an improvement, in the first rank, of more than 10% compared to the other multi-shot descriptors results with CAVIAR4REID dataset.

Table 2. Comparison between the recognition rate in the first rank of GL-VC and other approaches on CAVIAR4REID dataset

Approaches	Recognition rate in the first rank (%)
AHPE [13]	10
CPS [19]	17
Local fisher [5]	36
MFA [10]	40.2
GL-VC (our approach)	**53.4**

These experimental results prove that using multiple human images from a video sequence, involving time parameter, Gabor and LBP features with other appearance features is more informative and relevant to describe a human and reduce the effect of several noise types. But, the challenge remains at the level of increasing the execution time related, firstly to the selection of features in the mapping function and secondly to the GOP size.

5 Conclusion

In this work, we have presented a new approach in multi-shot human re-id. We have introduced a framework which generates and pre-treats the image sequences to apply S-T approaches on human re-id process. Then we have developed the GL-VC descriptor which provides extracting correlation between frames and modeling capabilities of people's motion with color and texture features. To improve this descriptor against the variations of illumination and view point and to render it more robust against noises, we have integrated the Gabor and LBP operators. In order to showcase the merits of our new approach, we have evaluated our GL-VC approach on the CAVIAR4REID multi-shot re-id dataset and demonstrated superior results when compared to the current state of the art. We have found that integrating appearance features with Gabor and LBP operators and finding its correlation with the motion modeling through a mapping function gives a considerable gain at the performances of the recognition process.

As a future work, we will focus on extending the proposed method so that it will be possible to take up the challenge of clothing similarity in appearance-based methods and the effect of background in human modeling.

Acknowledgements. This 'Mobidoc' research was achieved through the partnership agreement 'Programme d'Appui au Système de Recherche et d'Innovation' (PASRI) between the Government of the Tunisian Republic (ANPR) and the European Union.

References

1. Tuzel, O., Porikli, F., Meer, P.: Region covariance: a fast descriptor for detection and classification. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006. LNCS, vol. 3952, pp. 589–600. Springer, Heidelberg (2006). doi:10.1007/11744047_45
2. Bak, S., Bremond, F.: Re-identification by covariance descriptors. In: Gong, S., Cristani, M., Yan, S., Loy, C.C. (eds.) Person Re-identification. Advances in Computer Vision and Pattern Recognition, pp. 71–91. Springer, Heidelberg (2014)
3. Hadjkacem, B., Ayedi, W., Snoussi, H., Abid, M.: A spatio-temporal covariance descriptor for person re-identification. In: Proceedings of the 15th International Conference on Intelligent Systems Design and Applications, pp. 618–622 (2015)
4. Ayedi, W., Snoussi, H., Abid, M.: A fast multi-scale covariance descriptor for object re-identification. Pattern Recogn. Lett. **13**(14), 1902–1907 (2012)
5. Pedagadi, S., Orwell, J., Velastin, S., Boghossian, B.: Local fisher discriminant analysis for pedestrian re-identification. In: CVPR, pp. 3318–3325 (2013)
6. Farenzena, M., Bazzani, L., Perina, A., Murino, V., Cristani, M.: Person re-identification by symmetry-driven accumulation of local features. In: CVPR, pp. 2360–2367 (2010)
7. Gheissari, N., Sebastian, T., Hartley, R.: Person re-identification using spatiotemporal appearance. In: IEEE Conference on Computer Vision and Pattern Recognition, vol. 2, pp. 1528–1535 (2006)
8. Hadjkacem, B., Ayedi, W., Abid, M., Snoussi, H.: LBP based spatio-temporal covariance descriptor for people re-identification. J. Inf. Assur. Secur. **11**(3), 126–134 (2016)
9. Dou, J., Li, J.: Robust human action recognition based on spatio-temporal descriptors and motion temporal templates. Optik **125**(7), 1891–1896 (2014)
10. Xiong, F., Gou, M., Camps, O., Sznaier, M.: Person re-identification using kernel-based metric learning Methods. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014. LNCS, vol. 8695, pp. 1–16. Springer, Heidelberg (2014). doi:10.1007/978-3-319-10584-0_1
11. Bilinski, P., Bremond, F.: Contextual statistics of space-time ordered features for human action recognition. In: AVSS (2012)
12. Abdelhedi, S., Wali, A., Alimi, A.M.: Fuzzy logic based human activity recognition in video surveillance applications. In: Abraham, A., Wegrzyn-Wolska, K., Hassanien, A.E., Snasel, V., Alimi, A.M. (eds.) Proceedings of the Second International Afro-European Conference for Industrial Advancement AECIA 2015. AISC, vol. 427, pp. 227–235. Springer, Heidelberg (2016). doi:10.1007/978-3-319-29504-6_23
13. Bazzani, L., Cristani, M., Perina, A., Murino, M.: Multiple-shot person re-identification by chromatic and epitomic analyses. Pattern Recogn. Lett. **33**, 898–903 (2012)
14. Liu, C., Wechsler, H.: Gabor feature based classification using the enhanced fisher linear discriminant model for face recognition. IEEE Trans. Image Process. **11**(4), 467–476 (2002)
15. Ojala, T., Pietikäinen, M., Harwood, D.: A comparative study of texture measures with classification based on feature distributions. Pattern Recogn. **29**, 51–59 (1996)
16. Ying, Z., Shutao, L.: Gabor-LBP based region covariance descriptor for person re-identification. In: International Conference on Image and Graphics (ICIG), pp. 368–371 (2011)

17. Bilinski, P., Bremond, F.: Video covariance matrix logarithm for human action recognition in videos. In: International Joint Conference on Artificial Intelligence (2015)
18. Bedagkar-Gala, A., Shah., S.K.: Multiple person reidentification using part based spatio-temporal color appearance model. In: ICCV Workshops (2011)
19. Cheng, D., Cristani, M., Stoppa, M., Bazzani, L., Murino, V.: Custom pictorial structures for re-identification. In: Proceedings of the British Machine Vision Conference, vol. 68, pp. 1–11 (2011)
20. Pang, Y., Yuan, Y., Li, X.: Gabor-based region covariance matrices for face recognition. IEEE Trans. Circuit Syst. Video Technol. **18**(7), 989–993 (2008)
21. Hadjkacem, B., Ayedi, W., Abid, M.: Accordion representation based multi-scale covariance descriptor for multi-shot person re-identification. In: Blanc-Talon, J., Distante, C., Philips, W., Popescu, D., Scheunders, P. (eds.) ACIVS 2016. LNCS, vol. 10016, pp. 297–310. Springer, Heidelberg (2016). doi:10.1007/978-3-319-48680-2_27

Enriching Trajectories with Semantic Data for a Deeper Analysis of Patterns Extracted

Sana Chakri(✉), Said Raghay, and Salah el hadaj

LAMAI Laboratory, Department of Applied Mathematics
and Computer Sciences, Cadi Ayyad University, Marrakech, Morocco
chakri.sana@gmail.com

Abstract. Geographical Information System (GIS) stores several types of data collected from several sources in varied format. Thus geo-databases generate day by day a huge volume of data from satellite images and mobile sensors like GPS, among these data we find in one hand spatial features and geographical data, and in other hand trajectories browsed by several moving objects in some period of time. Merging these types of data leads to produce semantic trajectory data. Enriching trajectories with semantic geographical information lead to facilitate queries, analysis, and mining of moving object data. Therefore applying mining techniques on semantic trajectories continue to proof a success stories in discovering useful and non-trivial behavioral patterns of moving objects. The objective of this paper is to envisage an overview of semantic trajectory knowledge discovery, and spatial data mining approaches for geographic information system. Based on analysis of various literatures, this paper proposes a concept of multi-layer system architecture for raw trajectory construction, trajectory enrichment, and semantic trajectory mining.

Keywords: Spatiotemporal data mining · Semantic enrichment process · Geographic information system · Semantic trajectory knowledge discovery · Extracting behavioral knowledge

1 Introduction

Geographic Information System (GIS) has emerged as an efficient discipline due to the development of communication technologies. Enormous quantity of data is generated in the form of image or fat files from sources like satellite imagery, mobile sensors, etc. Understanding information stored in these databases requires computational analysis and modeling techniques. Spatial and spatiotemporal data mining has emerged as a new area of research for analysis of data with respect to spatial/temporal relations and eventually contextual data related. In this field, the comprehension of phenomena related to movement of objects like people, vehicles, or even animals which traverse across the geographical region has always been a key issue in many areas of scientific investigation. Although the information encapsulated is often rich when considering the semantics and contextual data associated to the underlying movement. So far an optimal result, a moving object trajectory should be enriched by spatial, temporal and semantic knowledge, this being denoted as a semantic enrichment process.

A. Abraham et al. (eds.), *Proceedings of the 16th International Conference on Hybrid Intelligent Systems (HIS 2016)*, Advances in Intelligent Systems and Computing 552,
DOI 10.1007/978-3-319-52941-7_21

Regarding the spatial dimension, a trajectory can be modeled as a series of episodes, which each episode is defined as a maximal homogeneous sub-sequence of a trajectory. This allows mapping a given trajectory to a series of spatial predicates whose semantics can be also enriched by additional application dependent criteria.

The research presented in this paper discuss a brief review of the structure of GIS and its fields of knowledge discovery, besides we will discuss semantic modeling approach for a specific type of spatiotemporal data which is semantic trajectories in order to enrich trajectory data with semantic data, going by several phases like data preprocessing, trajectory segmentation and semantic enrichment. Then apply mining algorithms in order to provide more meaningful patterns about moving objects trajectories.

The rest of the paper will be structured as follow: in Sect. 2 we will present the related work, in Sect. 3 we will give a brief overview of GIS functionalities and his integrated knowledge discovery and spatiotemporal data mining, Sect. 4 will provide with a multi-layer system for trajectory enrichment, in which we will discuss the flow to construction semantic trajectories then applying mining algorithms for extracting more meaningful patterns, Sect. 5 illustrate a case of study, and in Sect. 5 we will discuss the work proposed and give some comparisons.

2 Related Work

Spatial feature extraction is one of the prospective areas that spatial data mining has come to develop; it poses the challenge to reveal meaningful information of geographic objects, with a particular interest in their relationships. Information about spatial features can be derived from single objects but most communally it is derived from the relationship between two or more objects, this type of spatial features is called relational features and it makes the main difference between relational and spatial knowledge discovery [1]. In literature, there are three types of relational features; (i) distance relations which are the most prominent relational features and can be computed using Euclidian or any other type of distance to compute it. (ii) Topological relations [2] which are invariant under homomorphism, and can be preserved if the considered objects are rotated, scaled or moved. (iii) Directional relations [3] which are defined over a reference system determined by two orthogonal axes, x and y. The extraction of spatial features from geographic data (such as topological, distance or directional) is the most effort- and time-consuming step in the whole discovery process. On the one side the user must choose the appropriate spatial and non-spatial features. On the other side the extraction process itself requires high computational costs, besides, spatial features can be extracted either in the data preprocessing or during the data mining task [4, 5]. But most approaches extract spatial features in data preprocessing [6]. Considering applications where there is very little or no semantics in spatial features analysis, experts found difficulties to interpret results from the user's point of view because patterns are purely geometrical, so they cannot discover semantic patterns which can be independent of spatial location. Thereby, adding semantics enhances the analysis of data and ease the discovery of semantically implicit behaviors [7]. Our approach is one of the methods that encourage the use of semantics data before applying mining algorithms, in order to give meaning to behaviors extracted.

3 GIS Knowledge Discovery

3.1 Geographic Information System (GIS)

Today, it is no longer easy to give a clear definition of GIS. The development of information and communication technologies in GIS domain has generated huge volume of data representing geographical and spatial information. Nowadays a geographic information system gives us the possibility to visualize, question, analyze, interpret and predicate data to understand relationships, patterns, and trends. GIS databases store different types of information received from several components connected with each other. According to [8] it can be viewed as collection of components such as data, software, hardware, procedures and methods used by people for data acquisition, analysis and decision making with respect to location for solving complex planning and management problems.

However, Real world data contains many different aspects, GIS present them by the concept of layers, which correspond to themes in the application; data is divided over thematic layers, each layer represents a common type of data and therefore the information in one layer is of a similar type. Layers are described by two types of data (called features); spatial data and attribute data (called also thematic or non-spatial data); spatial data are stored in form of vector or raster images [9, 10]. Vector is a representation of the world using points, lines, and polygons. It is useful for storing data that has discrete boundaries, such as country borders, land parcels, and streets, while raster is a representation of the world as a surface of regular grid of cells. It is more useful for storing data that varies continuously, as in an aerial photograph, a satellite image, a surface of chemical concentrations, or an elevation surface.

Figure 1 shows The GIS database as a container used to hold GIS features.

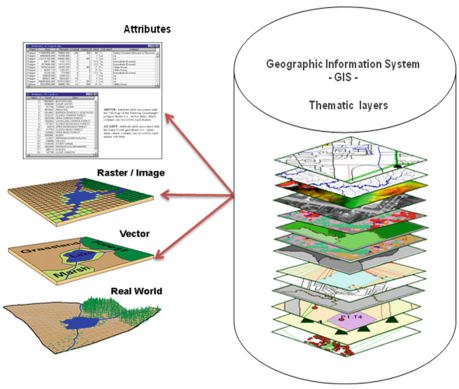


Fig. 1. GIS database

Analysis of data deposited in GIS has gained importance in domains related to knowledge management; recent use of spatial databases has lead to study Spatial and spatiotemporal Knowledge Discovery. However, GIS databases analysis isn't easy due to the varied formats, representation and data sources; mining spatial data from

geo-spatial datasets requires in somehow the help of domain experts to relate and understand spatial and non-spatial data, also it needs an understanding of information represented in image files (raster data), besides it needs interaction between spatial attributes and non-spatial attributes for an eventual selection and transformation.

3.2 Knowledge Discovery for GIS

The majorities of available GIS provide functionalities to manipulate and save data, but depend on the user's capacity for preparatory data analysis. The combination of data mining methods and GIS came to facilitate data analysis, and allows for an efficient execution of data mining algorithms.

Knowledge discovery in spatial databases refers to the extraction of implicit knowledge or other patterns that are not explicitly stored in database. A large amount of spatial data has been collected in various applications. The collected data is huge in such a way that it's far of human knowledge to analyze it, new and efficient methods are needed to discover knowledge from large spatial databases. From now on spatial knowledge discovery has emerged as an active research field in GIS, focusing on the development of theory, methodology, and practice for the extraction of useful information and knowledge from massive and complex spatial databases. Its main goal is analyzing large and complex spatial data which are multidimensional and auto correlated. To our best knowledge, there are a few software systems that join the data mining techniques and GIS functionalities, we can mention: GeoMiner [11], MoveMine [12], SPIN! [13, 14], INGENS [15] and WEKA-GDPM [16].

4 Semantic Trajectory Knowledge Discovery

First of all, it is important to notice the difference between spatial, geographic, and spatiotemporal object. A spatial object is an object that has location and geometry, 'Geographic' refers to the specific case where the spatial object is geo-referenced. Thus, the spatiotemporal object can be defined as an object that has at least one spatial and one temporal property. The spatial properties are location and geometry of the object. The temporal property is timestamp or time interval for which the object is valid. The spatiotemporal object usually contains spatial, temporal and thematic or non-spatial attributes. Examples of such objects are moving car, pedestrian, migration of birds or fish, forest fire, and earth quake. Spatiotemporal data sets essentially capture changing values of spatial and thematic attributes over a period of time, and structured it on trajectories.

Trajectory data play a fundamental role to a huge number of applications, such as transportation management, urban planning, tourism and animal migration. This type of data is normally obtained from mobile devices that capture the position of an object and his time interval, and it is available for use in the format of raw trajectories which often gives a few geometric behaviors. Semantic trajectories however, have more meaningful data; it is a growing trend that has recently emerged in geographic information science and spatiotemporal knowledge discovery. Their patterns are independent of spatial coordinates, and can be located in sparse regions and may not have

geometric similarity, from now on semantics plays an essential role in several applications, it is becoming an important research issue in GIS (Geographic Information Systems), however Complex technical operations and data functions are necessary to add semantics to trajectories in order to facilitate their analysis and knowledge extraction from the user's point of view [13, 14].

Figure 2 provide a multi-layer system for trajectory enrichment process, in which the main objective is constructing semantic trajectories from raw GPS feeds to prepare them for eventual use in spatial data mining.

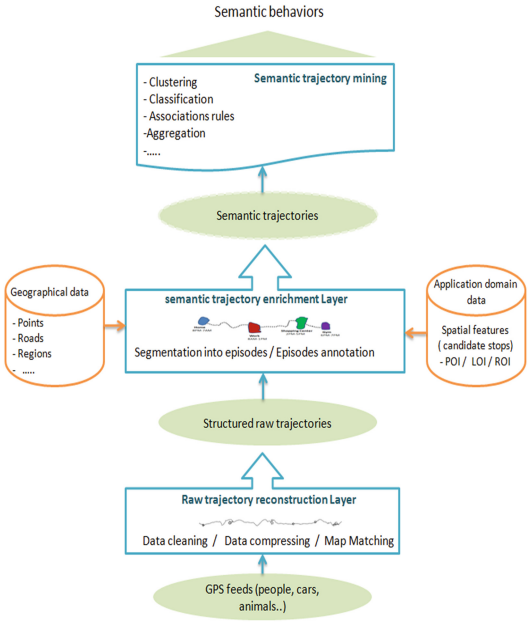


Fig. 2. Multi-layer system for trajectory enrichment

4.1 Raw Trajectory Reconstruction Layer

Raw data modeling is the first Layer of semantic trajectory process. It is the part of constructing and structuring data to support the enrichment of trajectories. In this phase there are three main levels that evolve as we progress from the initial requirements to the construction of structured trajectories; (1) Data cleaning, (2) Trajectory compression, (3) Map-matching; in which the global aim is to turn the imperfect raw movement data into a trajectory dataset that is correct and manageable. In other words we try to construct trajectories that are: clean (i.e., without/less noise), Accurate (i.e., map-matched) and compressed as much as possible.

4.2 Semantic Trajectory Enrichment Layer

After constructing raw data, and to prepare trajectories for data mining, the mid step is to add semantics to these trajectories, adding semantics is achieved by linking the spatiotemporal units with semantic knowledge from the geographic data in one hand and application domain data in other hand, to facilitate the user's task to analyze and interpret the knowledge in mining steps.

4.2.1 Segmentation into Episodes

A way of segmenting a trajectory is to split its path into periods of time when the object is considered as stationary (stops), and periods where the object is indeed moving (moves). According to this model, stops are the important places of a trajectory where the moving object has stayed for a period of time, while moves are the sub-trajectories between stops. Thus, semantic trajectory: is a sequence of alternating stops and move episodes, always starting with a move and having zero or more stops. Notice that trajectories can be segmented with criteria other than stops and moves like the means of transportation used by the moving object, or roads name; this knowledge is useful for all planning applications in city transportation management [15].

4.2.2 Episodes Annotation

After segmenting trajectories, the segmented episodes need to be annotated in order to be more enriched and to give meaning to the semantic trajectories, Usually each episode is annotated with its defining annotation: for instance, stop and move episodes are annotated as "stop" or "move", transportation mode episodes are annotated as "walk", "bus", "car", etc. Points, lines, or regions of interest influence episodes are annotated with the identifier of the corresponding POI, LOI, or ROI. Episodes may also bear more than one annotation to fulfill application requirements, for instance stop and move episodes may be additionally annotated with the corresponding POI, LOI, or ROI where the moving object stopped. Its type (e.g., home, work-place, restaurant), or even the activities pursued during the stop and move. Roughly, very few researches had focused on annotating moves, while theirs is an important part of research for understanding stops because stopping generally means that there is something of interest to do there.

4.3 Semantic Trajectory Mining

In every application domain, using contextual and semantic information enriches the meaning of the extracted information, it can help to understand the behaviors and reveal behaviors that would be difficult to identify without using semantics. Knowledge discovery from trajectories aims essentially at identifying behaviors in two categories: unknown behaviors or specific behaviors. Techniques used to identify behaviors are clustering trajectories, classifying trajectories into predefined classes (fishing ship trajectories), discovering common sequences of elementary movements (going from home to work then to shopping center) and identifying trajectories of objects whose movement is somehow related to each other, in particular, objects moving together (flocks of animals). For knowledge discovery in semantic trajectories, the integration with contextual geographic information plays an essential role towards the discovery of

easily understandable behaviors. Considering the semantics of space and the semantics of time, it gives more meaning to a behavior; otherwise it is important to observe that the integration of trajectories with contextual and semantic, spatial and temporal information is vital for the discovery of meaningful behaviors, because the majority of behaviors extracted from semantic trajectories cannot be obtained from raw data only. Once the trajectories are built, enriched, well clustered and eventually classified, users at this level can extract meaningful knowledge and analyze behaviors of their moving objects. Researches lead to several kinds of semantic behaviors that can be mined by using semantic trajectories, the most popular is the sequential pattern (visiting sequences of places), the frequent Pattern (visiting the same places with a certain frequency), the associated pattern (visiting associations of places for instance trajectories that visit hotels do also visit touristic places), the outlier pattern (trajectories that have different behavior in relation to other trajectories), etc. [16, 17].

5 Case of Study

Let's consider a dataset of citizens GPS feeds represented as sample points. The first step is modeling raw trajectory, in this step we construct a healthy raw trajectories from GPS feed, in our case study we use cleaning and map-matching process to minimize the noise and to have a sound set of trajectories, we don't need to compress our trajectory because we suppose that GPS feeds are regularly captured over some period of time, for this step. Besides the trajectory data of citizens, there is a need of geographic information of the city. So after configuring and cleaning trajectory datasets, now we can add semantics to trajectories. Thereafter, we will exploit the semantic trajectory modeling and mining steps, First the raw trajectories will be segmented and annotated then clustered, we do that by using the concept of stops and moves and the algorithm IB-SMoT, Notice that the segmentation and annotation of trajectories in the toolkit Weka-STPM are provided at the same time when we enrich the trajectories in order to facilitate the procedure to users. To enrich our trajectories we need some geographic data, so besides the trajectory data of citizens, there is also contextual data of the city. Let us suppose that for this application, the interesting feature types (candidate stops) include all types of Markets that exist in Marrakesh city. Merging raw trajectories with contextual data lead to produce semantic trajectories in two relations Stop and Move. Stops and moves can be computed on the spatial granularity of instance or type depending of the application needs; in this experiment we tend to compute stops on granularities of type since that what interested us is the type of the store frequenting by citizens, not essentially its name. Table 3 shows a part of a stop dataset with spatial granularities of type after applying the algorithm IB-SMoT which merges between raw trajectories and candidate stops already existed in the dataset.

Once the stops and moves are generated, which is the phase when semantics are added to trajectories and when time is most consumed between all steps of the knowledge discovery process, the data are ready for multiple-level mining. Furthermore, we specify what should be considered in the mining step and manage the aggregation in higher granularity levels (generalization) in order to apply mining algorithms for generating several types of patterns. Notice that generating stops and

moves from trajectories is normally realized for once, while data mining algorithms may be applied to these data several times until extracting meaningful patterns. According to the application domain and the user objectives, different data mining algorithms can be applied to the dataset of stops and moves; the most important are frequent patterns, association rules, and sequential patterns. For the rest of this experimental study we lead to extract some frequent patterns to analyze citizens' behavior regarding to different types of markets that exist in the city of Marrakesh. After adding semantics to trajectories with the method CBSMoT, we transformed the data into different space and time granularities and then applied the algorithm for generating frequent patterns; at this level we can use the semantic trajectory data mining query language (ST-DMQL) to specify semantic enrichment of trajectories with contextual domain information. Figure 3 shows the result of frequent pattern mining process on semantic trajectories considering two different granularities for the time dimension: weekdays and weekends [18].

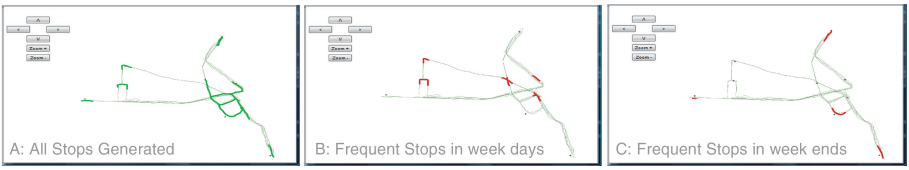


Fig. 3. Patterns extracted from semantic trajectories using IB-SMoT

6 Discussions and Comparisons

In Fig. 3 we have three results: (A) represent all stops computed by IB-SMoT, (B) shows stops computed in weekdays, these stops are located essentially near to small and medium types of Market. And (C) shows stops computed in weekends which report generally, besides than super Markets to hyper Markets and Malls. The information extracted by this pattern show that Marrakesh citizens frequent the small stores more than big surfaces, besides those who frequent big surfaces spend more time than those visiting super markets or small department stores, otherwise there is a relationship between the size of the building, and how long people attending it and spend time in it.

We can say that Behaviors extracted from semantic trajectories cannot be obtained from raw data only. According to the kind of knowledge the user is interested in, different data mining algorithms can be applied to the dataset of stops and moves. And without using such kind of conceptualization it would be hard to understand moving behavior in trajectories. In comparison with geometrical methods which are the traditional approaches, it is more significant to say that the object O has gone from cinema to the hotel passing by the supermarket, than saying that the object O has gone from Point C to point H stopping at an unknown point S. An example which expresses such necessity could be the following: (Begin, home, 8:00) → (move, road, 8:00–9:00, on-taxi) → (stop, office, 9:00–12:00, work) → (move, road, 12:00–12:30, walking) → (stop, restaurant, 12:30–13:30, lunching) → (move, road, 13:30–14:00, walking) → (stop, office, 14–18, work) → (move, road, 18–18:30, on-taxi) → (End, home, 18:30).

7 Conclusion

In this paper we have shown that trajectory knowledge discovery depends directly on the application domain, and therefore there is a need to integrate geographic information into trajectories in order to create semantic trajectories before extracting meaningful patterns. We have presented for that an overview for spatial data mining in GIS; afterwards we have presented a multi-layer system for trajectory enrichment to explain the semantic trajectory knowledge process, in which we present the main phases that ensure complete acquisition of semantic trajectories in order to extract realistic behaviors according the application point of view. The proposed multi-layer system process is general enough to cover different application domains. The future ongoing work is the study and the exploitation of the clustering techniques to find unknown stops that can reveal more important knowledge.

References

1. Zhixian, Y.: Semantic trajectories: computing and understanding mobility data. Ph.D. dissertation, Swiss Federal Institute of Technology, Information and Communication Department, Lausanne (2011)
2. Ilarri, S., Stojanovic, D., Ray, C.: Semantic management of moving objects: a vision towards smart mobility. Expert Syst. Appl. **42**(3), 1418–1435 (2015)
3. Panagiotakis, C., Pelekis, N., Kopanakis, I., Ramasso, E., Theodoridis, Y.: Segmentation and sampling of moving object trajectories based on representativeness. IEEE Trans. Knowl. Data Eng. **24**(7), 1328–1343 (2012)
4. Shekhar, S., Jiang, Z., Ali, R.Y., Eftelioglu, E., Tang, X., Gunturi, V.M.V., Zhou, X.: Spatiotemporal data mining: a computational perspective. ISPRS Int. J. Geo-Inf. **4**, 2306–2338 (2015)
5. Eldawy, A., Mokbel, M.F.: SpatialHadoop: a MapReduce framework for spatial data. In: Proceedings of the IEEE International Conference on Data Engineering (ICDE 2015), Seoul, Korea, 13–17 April 2015
6. Alvares, L., Palma, A., Oliveira, G., Bogorny, V.: Weka-STPM: from trajectory samples to semantic trajectories. In: Proceedings of the Workshop Open Source Code, vol. 1 (2010)
7. Albanna, B.H., Moawad, I.F., Moussa, S.M., Sakr, M.A.: Semantic trajectories: a survey from modeling to application. In: Popovich, V., Claramunt, C., Schrenk, M., Korolenko, K., Gensel, J. (eds.) Information Fusion and Geographic Information Systems (IF&GIS 2015). Lecture Notes in Geoinformation and Cartography, pp. 59–76. Springer, Heidelberg (2015)
8. Hastings, D.A.: Geographic Information Systems: A Tool for Geoscience Analysis and Interpretation (1992)
9. Rigaux, P., Scholl, M., Voisard, A.: Spatial Databases: With Application to GIS. Morgan Kaufmann, Burlington (2001)
10. Burrough, P.A., McDonnell, R.A.: Principles of Geographical Information Systems. Oxford University Press, Oxford (2000)
11. Han, J., Koperski, K., Stefanovic, N.: GeoMiner: a system prototype for spatial data mining. In: Proceedings of the International Conference on Management of Data (SIGMOD 1997), pp. 553–556. ACM (1997)

12. Li, Z., Ji, M., Lee, J., Tang, L., Yu, Y., Han, J., Kays, R.: MoveMine. In: Proceedings of the 2010 International Conference on Management of Data, SIGMOD 2010, p. 1203 (2010)
13. May, M., Savinov, A.: SPIN!-an enterprise architecture for spatial data mining. In: Palade, V., Howlett, R.J., Jain, L. (eds.) KES 2003. LNCS, vol. 2773, pp. 510–517. Springer, Heidelberg (2003). doi:10.1007/978-3-540-45224-9_70
14. SPIN! Spatial mining for public data of interest (2007). http://www.ais.fraunhofer.de/KD/SPIN/
15. Malerba, D., Esposito, F., Lanza, A., Lisi, F.A., Appice, A.: Empowering a GIS with inductive learning capabilities: the case of INGENS. J. Comput. Environ. Urban Syst. **27**(3), 265–281 (2003)
16. Bogorny, V., Palma, A.T., Engel, P., Alvares, L.O.C.: Weka-GDPM - integrating classical data mining toolkit to geographic information systems. In: SBBD Workshop em Algoritmos e Aplicações de Mineração de Dados (WAAMD 2006) (2006)
17. Cheng, T., Haworth, J., Anbaroglu, B., Tanaksaranond, G., Wang, J.: Spatiotemporal data mining. In: Nijkamp, P., Fischer, M.M. (eds.) Handbook of Regional Science, pp. 1173–1193. Springer, Heidelberg (2014)
18. Chakri, S., Raghay, S., El Hadaj, S.: Modeling, mining, and analyzing semantic trajectories: the process to extract meaningful behaviors of moving objects. Int. J. Comput. Appl. **124**(8), 15–21 (2015). Published by Foundation of Computer Science (FCS), NY, USA

Diagnosis of Alzheimer Disease from MRI Images of the Brain Throughout Time

Amira Ben Rabeh[1,2(✉)], Faouzi Benzarti[1,2], and Hamid Amiri[1,2]

[1] LR-11-ES17 Signal, Images et Technologies de l'Information (LR-SITI-ENIT), Tunis Le Belvédère, Tunisie
amira.benrabeh@gmail.com, benzartif@yahoo.fr, hamidlamiri@yahoo.fr
[2] Université de Tunis El Manar Ecole Nationale d'Ingénieur de Tunis, Tunis, Tunisia

Abstract. In this paper, we will present the system that we designed to track The Alzheimer disease's evolution throughout time using Magnetic Resonance Imaging (MRI) images. The AD makes visible changes in brain structures. We aim to identify the patient category as AD or Normal Control (NC) subject. The paper's contribution relies on realizing a method for longitudinal monitoring of a subject. Our method contains two parts: the first step allows the analysis of two MRI of the same patient in two different times to determine changes in the hippocampus texture descriptors, which are used to move to the second step which is the classification using the SVM method (Support Vector machine) based on a preliminary phase i.e. the learning phase.

Keywords: Alzheimer Diseases (AD) · Magnetic Resonance Imaging (MRI) · Normal Control (NC) · Support Vector machine (SVM)

1 Introduction

The Alzheimer disease is a neurodegenerative disease. It is the leading cause of heavy dependence of the elderly;It starts before the stage of dementia by the appearance of cognitive disorders which are variously associated, and by possible behavioral or personality troubles. The disease progresses over several years with the emergence of a progressive dependence impact on activities of daily living (dressing, feeding, movement) [1–6].

Alzheimer's disease (AD) is a major common health problem, charachterized by the lateness of the clinical diagnosis and the absence of treatment. Early diagnosis of this disease would be a major benefit medically, socially and economically. Furthermore, the development of new therapeutics requires a better understanding of the pathophysiological mechanisms underlying this disease. To better understand these mechanisms, we must have an integrated view of the different manifestations of AD (biological and cognitive) by combining complementary techniques, and studying their hierarchy of appearance through a longitudinal approach [7, 8].

AD is associated with cognitive changes mainly characterized by deficits in episodic memory tests, verbal fluency and executive functions, a major atropie of the

A. Abraham et al. (eds.), *Proceedings of the 16th International Conference on Hybrid Intelligent Systems (HIS 2016)*, Advances in Intelligent Systems and Computing 552,
DOI 10.1007/978-3-319-52941-7_22

hippocampal region and hypometabolism measured FDGTEP (or activation defect measured fMRI) of the posterior cingulate and temporoparietal regions. Potential biomarkers, tissue plasminogen activator (tPA), involved in synaptic plasticity phenomena, are a promising path. In terms of early diagnosis, if the 18 FDG-PET seems to be the most effective marker, its limited access compared to other techniques (MRI, neuropsychological, biological) justifies further research in this area, using new approaches (biomarkers, original neuropsychological testing, fMRI resting, activation fMRI), or original analysis techniques (furrows analyzes using data from IRMa) but also by combining different markers [9].

In the field of medicine, medical imaging may be defined as the process used to view inside the human body in vivo. It is a crucial tool for the physician to understand and diagnose the disease. Indeed, thanks to a preoperative knowledge of the patient's internal anatomy, the physician is able to establish a better diagnosis and better plan the therapeutic and surgical procedures.

However, Though the acquisition techniques are evolving daily, objective and quantitative interpretation of medical images is still a difficult task and a topic of strong current research, particularly, the segmentation step which is essential in the processing chain and analysis of medical images. It is a low-level treatment that aims to locate, in an image, the pixel sets (voxels) belonging to the same anatomical structures. This is what makes explicit the different regions of interest as well as the pathological aspects of the structures in the image. In addition, the result of segmentation can be the starting point for other medical imaging applications such as registration, reconstruction, visualization or analysis of movement [10–12].

Compared to other areas, this segmentation step is particularly difficult because of the artifacts associated with different acquisition systems as well as the nature of medical images: the low contrast and low resolution, lack of visible contours, the intensity and the similarity of anatomical structures, the heterogeneity of intensity within the same tissue, etc. To these properties, we add variability and complexity of the structures to be segmented. All these constraints make the segmentation difficult if based only on "low-level" features of the image, since the intensity of a pixel does not judge with certainty, belonging to a structure [13, 14]. In this context is our work, we will present a longitudinal tracking system. The paper is organized in this way, in the second part we will talk about our Computer Assisted Diagnosis. In the third part, we will present some results and the discussion section.

2 Proposed Application

Our proposed method contains two parts: Segmentation and classification. For the step of segmentation, we extract the ROI Region of Interest of the Hippocampus. We calculate the texture features of the hippocampus from the first IRM and the second MRI. We determine the variation of these Features.

After that we pass to the classification step. For the classification, we used the SVM (Support vector Machine) method. The step of classification is based on a training step, the figure in below presents the detail of the training step (Fig. 1).

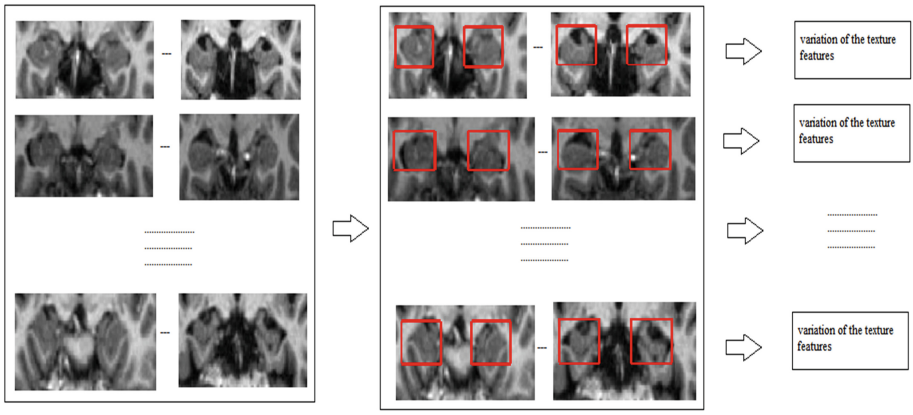


Fig. 1. Figure present the Training step

Training: For the training, we have 400 subjects: 200 Normal Subjects and 200 Alzheimer Subjects, for some subject there are two MRI along time (6 months) for the same patient. We analyze the two MRI and we extract the hippocampus after that we determine the variation of the vector of texture features.

The figure in below presents the principal model for our method (Fig. 2):

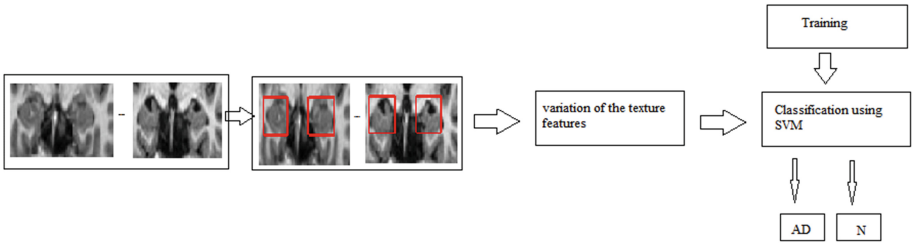


Fig. 2. Proposed Method

2.1 Segmentation

Our approach is applied to reference MRI based OASIS (Open Access Series of Imaging Studies) DATABASE. The images are a grayscale and size of 500 * 500 pixels. These images are made by experts. We extract the ROI (Region Of Interet) of the Hippocampus in the step of segmentation.

2.2 Extraction of Texture Features

The texture is a characteristic for the description of the visual content. It provides a description of the local structure as well as information on the random field that controls the small-scale details. Although several techniques have been proposed to

characterize any texture spring as a universal descriptor: each has advantages and disadvantages. We implemented the method of co-occurrence matrix grayscale to extract textures indexes because it is widely used in image search, and generally gives good results.

The co-occurrence matrix grayscale (GLCM) was proposed by Haralick et al. [15] In 1973.

This approach is based on the joint probability distribution of pixels in the image [16, 17].

2.3 Classification Using SVM (Support Vector Machine)

The support vector machine (Support Vector Machine, SVM) also called separators wide margin are supervised learning techniques to solve classification problems. The support vector machines uses the concepts of the theory of statistical learning and terminals theory Vapnik and Chervonenkis [18].

This technique is a method of classification to two classes which attempts to separate the positive examples from negative examples in all examples. The method then uses the hyperplane that separates the positive examples of negative examples, ensuring that the margin between the nearest positive and negative is maximum. This ensures a generalization of the principle as new examples will not be too similar to those used to find the hyperplane but be located on one side or the other of the border. The advantage of this method is the selection of support vectors that represent the discriminant vectors by which is determined the hyperplane. The examples used during the search of the hyperplane are no longer useful and only these support vectors are used to assign a new case, which can be considered an advantage for this method [19, 20].

SVM techniques (non-linear) use an implicit function Φ transforming the input space XC Rd in a Hilbert space (H, <.,.> of larger size. Learning is then made from the model $(\Phi(X), Y)$ in the space H, larger dimension certainly, but where it is hoped that the data is "more linearly separable." from a practical standpoint, it should be noted that the calculation of projections $\Phi(X)$ is not used in the method, only the scalar products $<\Phi(x)\Phi(x)>, (x, x') \in X2$ are required. However, these are given by a kernel K via the relation ("kernel trick")

$$K(x,x') = <\Phi(x), \Phi(x')> \tag{1}$$

The method therefore requires to select a core (as well as other parameters). Possible choices include in particular: [21, 22]

Radial Gaussian kernel (Gaussian RBF):

$$K(x,x') = \exp(- \sigma\|x - x'\|^2) \tag{2}$$

A SVM classifier is of the form:

$$C(X) = \text{sign}(< w, \Phi(x) > + b) \tag{3}$$

where w ε H and R ε b are parameters adjusted during the learning phase to
From a sample of examples {(xi, yi): 1 <= i <= n}.

$$\text{Minimize } \frac{1}{2}\|w\|^2 + \frac{C}{m}\sum_{i=1}^{n} E_i \tag{4}$$

Under the constraints: ∀ i ϵ {i,….,n},

$$E_i > = 0 \text{ et } yi\,(< w, \Phi(x) > + b) > = 1 - E_i \tag{5}$$

We can show that the solution w can be expressed as follows:

$$w = \sum_{i=1}^{n} a_i y_i \Phi(x_i) \tag{6}$$

3 Result and Discussion

The goal of our method is to analyze both MRI of the same patient, taken at two different times to detect that the patient is normal or Alzheimer's disease. We pass by the segmentation step of the two MRI to extract the both hippocampus after that we extract the texture descriptors. We search the variation vector of descriptors from the two MRI. To make the final decision, we pass by the classification stage using the classification method supervised SVM (Support Vector Machine). At the end, we get the patient is classified normal or Alzheimer's disease. It is sure that the classification phase, based on a learning phase to collect more samples, and this phase is based on a labeling step to secure the two ROIs: Hippocampus left and Hippocampus right.

We test our method with 50 subjects: 25 Normal Control, 25 Alzheimer diseases. We present two Subjects: subject1 Normal Subject and Subject 2 Alzheimer Patient.

Normal Subject (Fig. 3):

Fig. 3. Segmentation of two IRM in different times of the Normal Subject

Alzheimer Subject (Fig. 4):

Fig. 4. Segmentation of two IRM in different times of the Alzheimer Subject

Normal

Analyzing the accuracy curve, we obtained 80% accuracy using a single descriptor same precision for 5 descriptors, 90% to 10 descriptors, 85% to 15 descriptors and finally 90% to 22 descriptors.

Alzheimer

The figure below shows the precision of curve in terms of texture descriptors for subjects suffering from Alzheimer (Fig. 5).

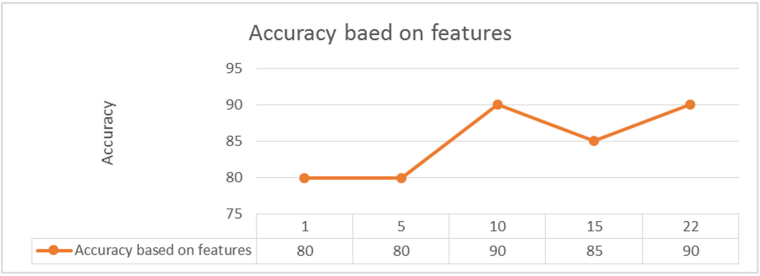


Fig. 5. Curve present the Accuracy based on features of the Normal Subjects

Analyzing the accuracy curve, 55% accuracy is achieved using a single descriptor, 65% for 5 descriptors, 80% to 10 descriptors, 85% to 15 descriptors and finally 85% to

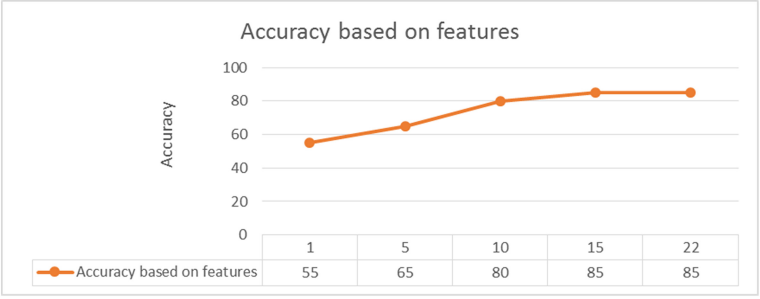


Fig. 6. Curve present the Accuracy based on features of the Alzheimer Subjects

22 descriptors. For both curves, we notice that the more one accumulates descriptors, the better the accuracy is. Also the choice of descriptors presents a good result in a diagnostic computer system (Fig. 6).

4 Conclusion

Alzheimer Disease is a progressive neurodegenerative disease and the most common cause of dementia in the elderly. Dementia refers to a state of progressive cognitive decline beyond the normal consequence of the expected aging. Since it is a progressive disease, the symptoms gradually worsen over time. This paper presented a new diagnostic method that can control two MRI of the same patient in two different periods. The change in the hippocampus is a good symptom to monitor patients. And our choice of the classification method using SVM (Support Vector Machine) provided us with the best results. We offer perspectives as adding other descriptors with the variation of the hippocampus to enhance our longitudinal tracking method.

References

1. Ryberg, C., Rostrup, E., Paulson, O.B., Barkhof, F., Scheltens, P., van Straaten, E.C., van der Flier, W.M., Fazekas, F., Schmidt, R., Ferro, J.M., Baezner, H., Erkinjuntti, T., Jokinen, H., Wahlund, L.O., Poggesi, A., Pantoni, L., Inzitari, D., Waldemar, G., LADIS study group: Corpus callosum atrophy as a predictor of age-related cognitive and motor impairment: a 3-year follow-up of the LADIS study cohort. J. Neurol. Sci. **307**, 100–105 (2011)
2. Cabezas, M., Oliver, A., Lladó, X., Freixenet, J., Cuadra, M.B.: A review of atlas-based segmentation for magnetic resonance brain images. Comput. Methods Programs Biomed. **104**, e158–e177 (2011)
3. Di Paola, M., Di Iulio, F., Cherubini, A., Blundo, C., Casini, A.R., Sancesario, G., Passafiume, D., Caltagirone, C., Spalletta, G.: When, where, and how the corpus callosum changes in MCI and AD: a multimodal MRI study. Neurology **74**, 1136–1142 (2010)
4. Di Paola, M., Luders, E., Di Iulio, F., Cherubini, A., Passafiume, D., Thompson, P.M., Caltagirone, C., Toga, A.W., Spalletta, G.: Callosal atrophy in mild cognitive impairment and Alzheimer's disease: different effects in different stages. Neuroimage **49**, 141–149 (2010)
5. Di Paola, M., Spalletta, G., Caltagirone, C.: In vivo structural neuroanatomy of corpus callosum in Alzheimer's disease and mild cognitive impairment using different MRI techniques: a review. J. Alzheimers Dis. **20**, 67–95 (2010)
6. Frederiksen, K.S., Garde, E., Skimminge, A., Ryberg, C., Rostrup, E., Baaré, W.F., Siebner, H.R., Hejl, A.M., Leffers, A.M., Waldemar, G.: Corpus callosum atrophy in patients with mild Alzheimer's disease. Neurodegener. Dis. **8**, 476–482 (2011)
7. Hampel, H., Teipel, S.J., Alexander, G.E., Horwitz, B., Teichberg, D., Schapiro, M.B., Rapoport, S.I.: Corpus callosum atrophy is a possible indicator of region- and cell type-specific neuronal degeneration in Alzheimer disease: a magnetic resonance imaging analysis. Arch. Neurol. **55**, 193–198 (1998)

8. Hampel, H., Teipel, S.J., Alexander, G.E., Horwitz, B., Teichberg, D., Schapiro, M.B., Rapoport, S.I.: Corpus callosum atrophy is a possible indicator of region- and cell type-specific neuronal degeneration in Alzheimer disease: a magnetic resonance imaging analysis. Arch. Neurol. **55**, 193–198 (1998)
9. Zhu, M., Gao, W., Wang, X., Shi, C., Lin, Z.: Progression of corpus callosum atrophy in early stage of Alzheimer's disease: MRI based study. Acad. Radiol. **19**, 512–517 (2012)
10. Chene, G., Beiser, A., Au, R., Preis, S.R., Wolf, P.A., Dufouil, C., et al.: Gender and incidence of dementia in the Framingham Heart Study from mid-adult life. Alzheimers Dement (2014). (Epub ahead of print)
11. Gilligan, A.M., Malone, D.C., Warholak, T.L., Armstrong, E.P.: Health disparities in cost of care in patients with Alzheimer's disease: an analysis across 4 state Medicaid populations. Am. J. Alzheimers Dis. Other Dement **28**(1), 84–92 (2013)
12. American Psychiatric Association: Diagnostic and Statistical Manual of Mental Disorders, 5th edn. American Psychiatric Publishing, Arlington (2013)
13. Villemagne, V.L., Burnham, S., Bourgeat, P., Brown, B., Ellis, K.A., Salvado, O., et al.: Amyloid deposition, neurodegeneration, and cognitive decline in sporadic Alzheimer's disease: a prospective cohort study. Lancet Neurol. **12**(4), 357–367 (2013)
14. Wilson, R.S., Boyle, P.A., Yu, L., Barnes, L.L., Schneider, J.A., Bennett, D.A.: Life-span cognitive activity, neuropathologic burden, and cognitive aging. Neurology **81**(4), 314–321 (2013)
15. Haralick, R.M.: Statistical and structural approches to texture. Proc. IEEE **67**(5), 786–804 (1979)
16. Haddon, J.F., Boyce, J.F.: Co-occurrence matrices for images analysis. IEEE Electron. Commun. Eng. J. **5**(2), 71–83 (1993)
17. Haralick, R.M., Shanmugam, K., Dinstein, I.: Textural features for images classification. IEEE Trans. Syst. Man Cybern. **SMC-3**(6), 610–621 (1973)
18. Vapnik, V.: Statistical Learning Theory. Wiley, NY (1998)
19. Vapnik, V.N.: The Nature of Statistical Learning Theory. Springer, New York (1995)
20. Bradley, P.S., Mangasarian, O.L.: Feature selection via concave minimization and support vector machines. In: Shavlik, J. (ed.) Machine Learning Proceedings of the Fifteenth International Conference (ICML 1998), pp. 82–90. Morgan Kaufmann, San Francisco, California (1998). ftp://ftp.cs.wisc.edu/math-prog/tech-reports/98-03.ps
21. Mangasarian, O.L.: Arbitrary-norm separating plane. Oper. Res. Letters **24**, 15–23 (1999). ftp://ftp.cs.wisc.edu/math-prog/tech-reports/97-07r.ps
22. Cervantes, J., Li, X., Yu, W., Li, K.: Support Vector Machine Classification For Large Data Sets via Minimum Enclosing Ball Clustering. Elsevier B. V., July 2007

Energy Aware Hybrid Scheme of Client-Server and Mobile Agent Models for Data Aggregation in Wireless Sensor Networks

Mohamed El Fissaoui[1(✉)], Abderrahim Beni-Hssane[1], and Mostafa Saadi[2]

[1] LAROSERI Laboratory, Computer Science Department, Sciences Faculty, Chouaib Doukkali University, El Jadida, Morocco
mohamed.el.fissaoui@gmail.com, abenihssane@yahoo.fr
[2] Departement Informatique et Telecoms Ecole Nationale des Sciences Appliquees (ENSA), Khouribga Universite Hassan 1er Settat, Settat, Morocco
saadi_mo@yahoo.fr

Abstract. In wireless sensor networks, the most commonly used model for data aggregation is client-server; where each node send its collected data to the sink. With the growing size of the network this model become inefficient, because of the huge amount of the data that should be processed to the sink, therefore mobile agent model present a good alternative to the traditional client-server model. In this model a mobile code migrate to the source nodes to collect data. It has been proven that planning itinerary for the mobile agent is a NP-Hard problem. In this paper, we present a hybrid scheme of client-server and mobile agent. At first we organize nodes in clusters, then select a node at the center as cluster head that will be responsible of clusters' data collection, after that we plan the itinerary among those cluster heads using minimum spanning tree, then dispatch the mobile agent to collect data from cluster heads. Simulations results show that our novel proposed scheme perform better in comparison to the existing schemes.

Keywords: Wireless sensor networks · Client-Server · Mobile agent · Itinerary planning · Minimum spanning tree · Data aggregation

1 Introduction

Wireless sensor networks have become one of the most recent topic of research [1]. Its use spread more and more in many fields and its applications have become divers such as flood detection, smart transportation, habitat monitoring, fire detection, air pollution monitoring, water quality monitoring, industrial monitoring, etc. Wireless sensor networks is composed of sensor nodes spatially distributed in monitoring area to collect the data and send it back to the processing element, where the data processing take place.

The most used model for data aggregation in wireless sensor networks (WSNs) [1] is client-server; where each node sends its collected data to the sink. But recently mobile agent model has been adopted by researchers, in this model instead of each node sends its collected data to the sink, the mobile code migrate to the source nodes to

A. Abraham et al. (eds.), *Proceedings of the 16th International Conference on Hybrid Intelligent Systems (HIS 2016)*, Advances in Intelligent Systems and Computing 552,
DOI 10.1007/978-3-319-52941-7_23

collect data. Mobile agent is a special kind of software entity that migrates among source nodes to collect data. This model has many features [2], which make it more suitable than the traditional model.

The most challenging issue with this model is itinerary planning; it has been proven that it's an NP-hard problem [3]. By planning mobile agent itinerary in an efficient way, the mobile agent can migrate among source nodes and collect data efficiently. Herein, we propose a novel hybrid scheme by combining the strengths of client-server and mobile agent models. With small number of source nodes client-server performs better than the mobile agent [4, 5]. But when the number of source nodes increase client-server model become inefficient because of the big amount of data that should be processed to the sink. In the other side, mobile agent model start to perform better [4, 5] because of its strategy to migrate from node to node to collect data instead of sending all data to the sink as in client-server model.

Taking in consideration the aforementioned issues, the main contribution of this paper are:

- Group source nodes in clusters and assign a node as cluster head
- Plan the itinerary among cluster heads using minimum spanning tree
- Dispatch mobile agent to collect data from cluster heads
- Compare and evaluate the proposed scheme with the existing schemes using Castalia Simulator [6]

The rest of this paper is organized as follows: in Sect. 2, we briefly describe related work. In Sect. 3, we describe the network model. In Sect. 4, we present our proposed scheme. In Sect. 5, we describe the simulation setup and discuss the performance analysis. Finally, we conclude this paper in Sect. 6.

2 Related Work

Recently several mobile agent based data aggregation approaches in wireless sensor networks have been proposed. In this section, we survey the literature related to these proposed approaches.

In [7] authors propose two heuristics algorithms; Local Closest First (LCF) and Global Closest First (GCF) to plan the itinerary of MA among source nodes to perform data fusion tasks. In LCF, the MA migration starts from the sink then the closest node to its current location and so on, and in GCF, the MA migrate to the next node closest to the sink. In LCF algorithm, the last source nodes in the MA are the nodes with long distance from the sink. Because LCF choose the next destination node among the source nodes based on its local location, not by looking at the global network distance matrix. Repetitive MA oscillations produced by GCF around the sink, result long itinerary and poor performance [7].

In [8], the authors propose the mobile agent based directed diffusion (MADD), although quite similar to LCF, MADD selects the farthest source node from the sink as the first source.

The authors in [9] propose a better scheme named itinerary energy minimum for first-source-selection (IEMF) algorithm, as well as the itinerary energy minimum

algorithm (IEMA), the iterative version of IEMF. IMEF denotes the importance of choosing the first visiting node. Based on this conclusion, it estimates energy costs of different alternatives of the first node and selects the solution with the minimum energy cost. In each iteration IEMA choose from the rest of source nodes, the source node with the optimal energy cost as the next node to visit.

In [3] authors proposed a Genetic Algorithm (GA). This algorithm provides superior performance than the LCF and GCF algorithms, but it implies a time-expensive optimal itinerary calculation, which is unacceptable for time-critical applications.

As it's obvious from the previous proposed algorithms, the mobile agent should visit all source nodes in the network to collect data which is not efficient; because when the mobile agent visit each source node, the size of mobile agent become bigger and bigger which affect the overall execution time and consume more energy. This why we propose a hybrid scheme of client server and mobile agent models, where the mobile agent visit only the cluster heads and not all source nodes in the network to collect data.

3 Network Model

We use the network model as shown in the Fig. 1; the sink is at the center of the monitoring area. The monitoring field is 500 × 500. Sensor nodes are distributed randomly in the monitoring area.

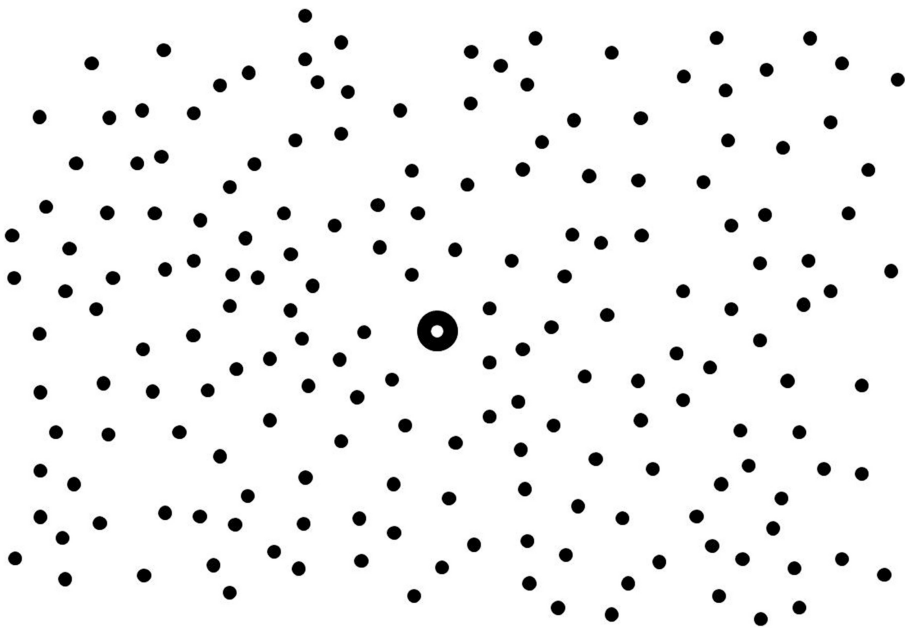


Fig. 1. Network model

4 Proposed Scheme

In this section, we present our novel energy aware hybrid scheme of client-server and mobile agent models for data aggregation in wireless sensor networks. The basic idea of our proposed scheme is to group source nodes in clusters and plan the itinerary among cluster heads, then dispatch the mobile agent to gather data from cluster heads.

At first, we group source nodes in clusters based on nodes density. In fact, at first we select a node as cluster head then include all nodes in its range in the cluster, we repeat this process until there is no node out of a cluster. At the end of this phase the network is organized in clusters.

After selecting cluster heads and constructing clusters, we use Minimum Spanning Tree (MST) algorithm to plan itinerary among cluster heads. To choose the best itinerary between two cluster heads, we calculate all possible paths and choose the path with the minimumm weight till the last node in the itinerary.

As shown in the algorithm, we start from the sink then choose the first cluster head with minimum weight from the sink as the next node and so on, till all cluster heads are included in the itinerary.

Itinerary planning among cluster heads based minimum spanning tree algorithm

```
TCH ← Sink
V   ← VCH
WHILE ∃ (u ∈ TCH, v ∈ V) DO
      Find min weight(u,v)
      TCH ← TCH + v
        V ← V - v
END WHILE
```

Finally, After organizing the network in clusters and plan the itinerary among those cluster heads, we dispatch the mobile agent to gather data from the cluster heads, at first the mobile agent send a notification to the source nodes in the same cluster notifying them to send the collected data to the cluster heads, and in its way back to the sink, the mobile agent start to collect data previously sent by source nodes from cluster heads.

5 Simulation Setup and Discuss the Performance Analysis

We use Castalia simulator [6] to implement our proposed scheme and compare it to the other existing schemes. Sensor nodes were randomly deployed in square monitoring area of 500 × 500 m, and varied from 200 to 800 nodes, also had the same transmission range and battery power, except for the sink, which is more powerful in terms of computation and battery power.

The rest of the simulation parameters are shown in Table 1.

We use energy consumption metric to evaluate our novel proposed scheme.

Figure 2 illustrate the comparison results of energy consumption per round of GCF, IEMF, MAAD, with our proposed scheme Energy aware hybrid scheme of client-server and mobile agent models (HCSMA), and with the increase of the network size from 200 to 600 nodes. As shown in Fig. 2, the energy consumption average per

Table 1. Simulation parameters

Heading level	Font size and style
Network transfer rate	250 Kbps
Monitoring field size	500 × 500 m
Number of nodes	[200, 800]
Node energy	18720 J
Energy consumed by MA execution	5nJ
MA processing delay	50 ms
MA instantiation delay	10 ms
Collected data size at each node	200 Bytes
MA code size	1024 Bytes

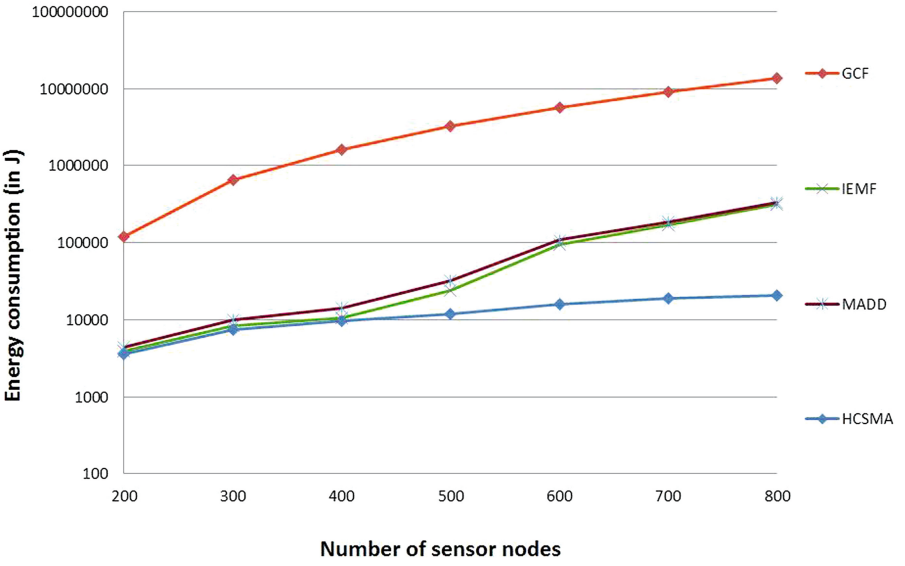


Fig. 2. Energy consumption per round

round is lower than all other protocols in the same scenarios. This is due to the strategy of visiting only the cluster head nodes to collect data that already sent by source nodes. GCF algorithm consume the biggest amount of energy in comparison to other algorithms, this is because of its poor method of visiting nodes that are far away from each other and the heavy weight of the mobile agent.

In the other side, with the network of 200,300,400 nodes IEMF and MADD and HCSMA consume almost the same amount of energy. But with the growing size of the network (500, 600, 700, 800) HCSMA consume less energy than IEMF and MADD, this is due to HCSMA strategy of visiting only cluster heads instead of visiting each node. By visiting each node as IEMF and MADD algorithms, the size of the mobile agent become bigger and consequently consumes more nodes energy. Also by visiting all the source nodes in the network, the length of the mobile agent itinerary became

longer, and as result the mobile agent heavy size affect the node energy' and network energy of IEMF and MADD dramatically as shown in the figure.

6 Conclusion

Client-server model for data aggregation in wireless sensor network is more efficient with network of small size, but with the growing size of the network this model become inefficient, because of the big amount of data that should be processed to the sink. In mobile agent model, the mobile agent migrates from node to node to collect data instead of sending the data to the sink at the same time. In our proposed scheme we combine the strengths of client-server and mobile agent models by grouping source nodes in clusters as client–server model and plan the itinerary among those clusters then send a mobile agent to collect data, this way the mobile agent collect data just from cluster heads not all source nodes.

Simulation result show that our novel proposed scheme consume less energy than the previous proposed schemes, this is due to the use of client–server model at first then select a nodes as cluster head that will be responsible of the collection of clusters' data, and at the end the sink dispatch the mobile agent to collect data from the cluster heads, when the mobile agent visit the cluster heads for the first time, it notify the cluster heads to send a message to the nodes in the same cluster to send the data to the cluster head, and in its way back to the sink, the mobile agent start to gather data from cluster heads.

References

1. Akyildiz, I.F., Su, W., Sankarasubramaniam, Y., Cayirci, E.: Wireless sensor networks: a survey. Comput. Netw. **38**(4), 393–422 (2002)
2. Lange, D.B., Oshima, M.: Seven good reasons for mobile agents. Commun. ACM **42**(3), 88–89 (1999). http://doi.acm.org/10.1145/295685.298136
3. Wu, Q., Rao, N.S.V., Barhen, J., Iyengar, S.S., Vaishnavi, V.K., Qi, H., Chakrabarty, K., Member, S., Member, S.: On computing mobile agent routes for data fusion in distributed sensor networks. IEEE Trans. Knowl. Data Eng. **16**, 740–753 (2004)
4. Qi, H., Xu, Y., Wang, X.: Mobile agent based collaborative signal and information processing in sensor networks. Proc. IEEE **91**, 1172–1183 (2003). Foster, I., Kesselman, C., Nick, J., Tuecke, S.: The physiology of the grid: an open grid services architecture for distributed systems integration. Technical report, Global Grid Forum (2002)
5. Boulis, A.: Castalia: simulator for wireless sensor networks and body area network (2012)
6. Qi, H., Wang, F.: Optimal itinerary analysis for mobile agents in and hoc wireless sensor networks (2001)
7. Chen, M., Kwon, T., Yuan, Y., Choi, Y., Leung, V.C.M.: Mobile agent-based directed diffusion in wireless sensor networks. Eurasip J. Appl. Signal Process. **2007**(1), 219–242 (2007)
8. Chen, M., Yang, L.T., Kwon, T., Zhou, L., Li, M.: Itinerary planning for energy efficient agent communications in wireless sensor networks. IEEE Trans. Veh. Technol. **60**, 3290–3299 (2011)
9. Wu, Q., Rao, N.S.V., Barhen, J., Iyengar, S.S., Vaishnavi, V.K., Qi, H., Chakrabarty, K., Member, S., Member, S.: On computing mobile agent routes for data fusion in distributed sensor networks. IEEE Trans. Knowl. Data Eng. **16**, 740–753 (2004)

3D CT Denoising by New Combination Between Nl-Mean Filter and Diffusion Tensor

Feriel Romdhane(✉), Faouzi Benzarti, and Hamid Amiri

Signal, Image and Information Technology Laboratory (SITI),
National Engineering School of Tunis, El Manar University, Tunis, Tunisia
ferielromdhane@yahoo.fr, benzartif@yahoo.fr,
hamidlamri@gmail.com

Abstract. The images acquired from different techniques of medical equipment are generally noisy data. The noise distribution in the CT image is modeled as a Gaussian distribution which appears in the images as a random fluctuation allowing to a misdiagnosis. So the denoising of the CT images is a challenge task in medical area. In this paper we propose a new denoised method based on combination between the Non local mean filter and the Diffusion Tensor for 3D Computed tomography scan data with a MAD estimator for gaussian noise. A quantitative measures was calculated and compared to other common denoising methods, improving the efficiently of our algorithm in term of removing noise and preserving significant details.

Keywords: Gaussian noise · CT-scan data · Non local mean filter · Diffusion tensor · MAD estimator

1 Introduction

The Computed Tomography (CT) is one of the commonly techniques in medicine which defines the normal and the abnormal structures in the body. It consists in measuring the X-ray absorption by the tissues and by the computer processing, scanning and finally reconstructing 2-dimensional (2D) images or 3-dimensional (3D) anatomical structures. The noise level in the CT images is related to the dose quantity, so further decreasing the dose more the images obtained are noisy, but otherwise large radiation increase the risk of cancer. In literature, several denoised methods for CT data have been proposed. We notice the traditional linear filters such as the median filter [1], statistical linear filtering in spatial domain made by Lee [2], the wavelet transform [3, 4], the filters based on the use of Partial Differential Equations (PDE) [5–11], the Non local mean (NL-mean) filter introduced by Buades [12–15]. In our previous paper we proposed a new method for denoising 3D Magnetic Resonance Images (MRI) where we combined between the NL-mean filter and the anisotropic diffusion tensor [16] with a Rician nose estimator. In this paper we apply our proposed algorithm [16] in CT scan data in order to generalize the algorithm on different types of medical images. And since the distribution of the noise on CT images assume to Gaussian distribution, we modify the estimator noise present in our previous paper by a median absolute deviation estimator for gaussian noise. Our paper is organized as follows: Sect. 2, we introduce the median absolute

A. Abraham et al. (eds.), *Proceedings of the 16th International Conference on Hybrid Intelligent Systems (HIS 2016)*, Advances in Intelligent Systems and Computing 552,
DOI 10.1007/978-3-319-52941-7_24

deviation estimator for Gaussian noise. Section 3 defines our proposed algorithms with new estimator. Section 4 presents results of denoising 3D CT images; and Sect. 5 concludes the work.

2 Median Absolute Deviation Estimator for Gaussian Noise

The Median Absolute Deviation (MAD) is a robust, simple and easy tool for noise estimation. For a batch of numbers $\{x_i,\ldots x_n\}$, the MAD is defined as follows [17]:

$$MAD = b\,\mathrm{med}_i(|x_i - \mathrm{med}_j(x_j)|) \tag{1}$$

Where $\mathrm{med}_i(.)$ is the median of all the numbers by arranging all the observations from lowest value to highest value and picking the middle one; when n is an odd number, the median is a simple middle value and in case of a even number the median is then defined as the average of numbers with ranks $(n/2)$ and $((n/2)+1)$, x_j the n original observations and b is a constant to make the estimator consistent for the parameter of interest, so in order to estimate the standard deviation σ for a Gaussian distribution, b is usually set to be equal to 1.4826.

3 Our Previous Work

In our previous paper [16], we have proposed a new method based on combination between two filters such as the NL-mean filter and the anisotropic diffusion tensor with an estimator noise Rician for denoising 3D MRI Magnetic resonance images. In this section we present at first a stat of art about the two combined filters then we identify our proposed algorithm based on the Gaussian noise estimator detailed in Sect. 2.

3.1 State of the Art

On the one hand, The NL-mean filter was introduced by Buades [12], it compute a weighted average of all pixels in the image, taking into account the similarity of the local neighborhoods. The restored intensity $NL(I)(x_i)$ of the voxel x_i is the weighted average of the intensities of all voxels in the image I:

$$Nl(I) = \sum_{x_j \in \Omega^3} w(x_i, x_j) I(x_j) \tag{2}$$

Where $w(x_i,x_j)$ is the weight assigned to the restoration of voxel x_i which evaluate the similarity between the intensity of patches N_i and N_j registered to voxels x_i and x_j based on the square Euclidean distance between patches intensity, and defined as:

$$w(x_i, x_j) = exp(-d_I^2(x_i, x_j)/h^2)/Z(x_i) \tag{3}$$

Where

$$d_I^2(x_i, x_j) = \left\| I(N(x_i)) - I(\mathrm{N}(x_j)) \right\|^2 \tag{4}$$

And

$$Z(x_i) = \sum_{x_j \in \Omega^3} \exp(\left\| I(N_i) - I(N_j) \right\|^2 / h^2) \tag{5}$$

Z is a normalization constant ensuring that $\sum w(x_i,x_j) = 1$, $h^2 = \beta^2\sigma^2$ is a smoothing parameter to control the decreasing of the exponential function for $\beta \in$ [0,8;1.2] and the noise power. On the other hand, the anisotropic diffusion is considered as a powerful tool in the medical image filtering by describing local variation present in images. Weickert [18] used a nonlinear partial differential equation (PDE) based on the diffusion tensor given by:

$$\partial I_t = div(D.\nabla I) \tag{6}$$

Where $D(.)$ is a symmetric positive semi-definite matrix constructed from a structure tensor, written in case 3D as follows:

$$D = \begin{bmatrix} a & d & e \\ d & b & f \\ e & f & c \end{bmatrix} \quad with\, D(i,j) = \sum_{n=1..3} \lambda_n v_{ni} v_{nj} \tag{7}$$

Where v_1, v_2 and v_3 the eigenvectors of the structure tensor and λ_1, λ_2 and λ_3 the diffusion function calculate to reflect the diffusion process of the intensity of the image around the neighboring voxels for a time t. Therefore Weickert proposed two functions: the edge enhancing diffusion (EED) and the coherence enhancing diffusion (CED) [11, 18].

3.2 Our Proposed Algorithm

In order to preserves more structures in medical images we proposed a new method based on a combination between two common filters such as NL-mean and the anisotropic diffusion tensor by modifying the weight average described Eq. (3) by the new one:

$$w(x_i, x_j) = \frac{1}{Z(x_i)} e^{-\left(\frac{d_I^2(x_i,x_j) + d_{ID}^2(x_i,x_j)}{h}\right)^2} \tag{8}$$

Where $d_I^2(x_i, x_j)$ the square Euclidean distance between patches intensity of the noisy image, $d_{ID}^2(x_i, x_j)$ is the square Euclidean distance between patches intensity of the denoised image by the anisotropic diffusion model with original 3D-CED function defined as follows:

$$\begin{array}{ll} \lambda_1 = \alpha & \\ \lambda_2 = \alpha & \\ \lambda_3 = 1 & if \quad \mu_2 = \mu_3 = 0 \\ \lambda_3 = \alpha + (1-\alpha)\, exp(-ln(2)\lambda_c^2/k) & else \end{array} \tag{9}$$

Where $k = (\mu_2/(\alpha + \mu_3))^4$, $\alpha = 0.001$ [19], λ_c is the CED contrast parameter and finally h is a new modified smoothing parameter described as follows:

$$h = \beta.\sigma_{MAD} \tag{10}$$

Where σ_{MAD} is a standard deviation estimated by MAD estimator defined in Sect. 2. Our proposed method depends not only on the similarity of the intensity grey level of patches but also on the structures information by the diffusion along the third direction with noise estimation in iteratively way.

4 Experimental Results

In order to prove the efficiency of our proposed algorithm, we use a synthetic 3D Shepp-Logan Phantom with volume size (128 × 128 × 102) which was created as a standard for computerized tomography (CT) image reconstruction simulations of the head (Fig. 1). We add a Gaussian white noise with different levels (Figs. 2 and 3) with a comparison to different common denoising method, we denote the Total Variation model, Wavelet NeighShrinkSure denoised method [20] and PFNLM method [21]. In addition, different quantitative measurements were calculated such as the root mean square error (RMSE), Peak Signal-to-Noise Ratio (PSNR) and the Structural Similarity Index (SSIM) [22] between simulated images and the ground-truth noise free image in order to evaluate the results of the denoising proceeding. We set the parameters of our method as follow according to [19, 21]: $\beta = 0.1$, $\alpha = 0.001$, $\rho = 15$ and $\lambda c = 20$.

Fig. 1. Four transversal slices from ground-truth noise free 3D Shepp-Logan Phantom.

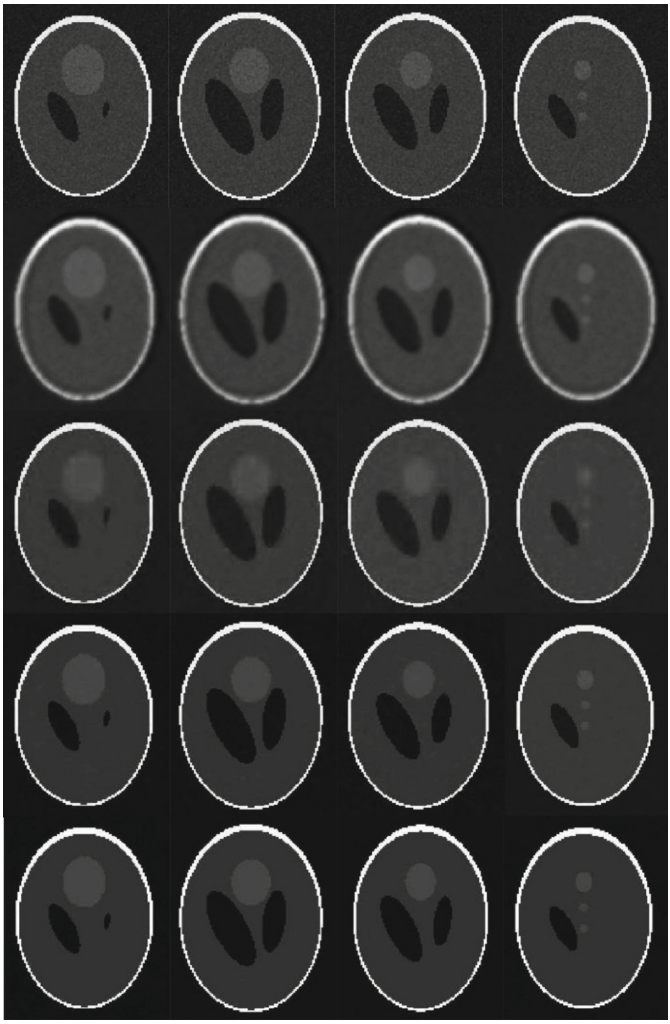


Fig. 2. Comparative results of denoising slices from 3D Shepp-Logan Phantom with 3% of add Gaussian White noise. From top to bottom are the results of: Noisy slices, Total Variation model (TV), Wavelet NeighShrinkSure denoised method, PFNLM method and our proposed method.

The Figs. 2 and 3 present the denoising results of denoising slices from 3D Shepp-Logan Phantom corrupted with 3% and 5% of add Gaussian White noise respectively. Visually the different methods have good performance against the additive noise and especially in low level rate. Moreover, the results show that our proposed method seems better in term of preserving edges and structures and removing noise.

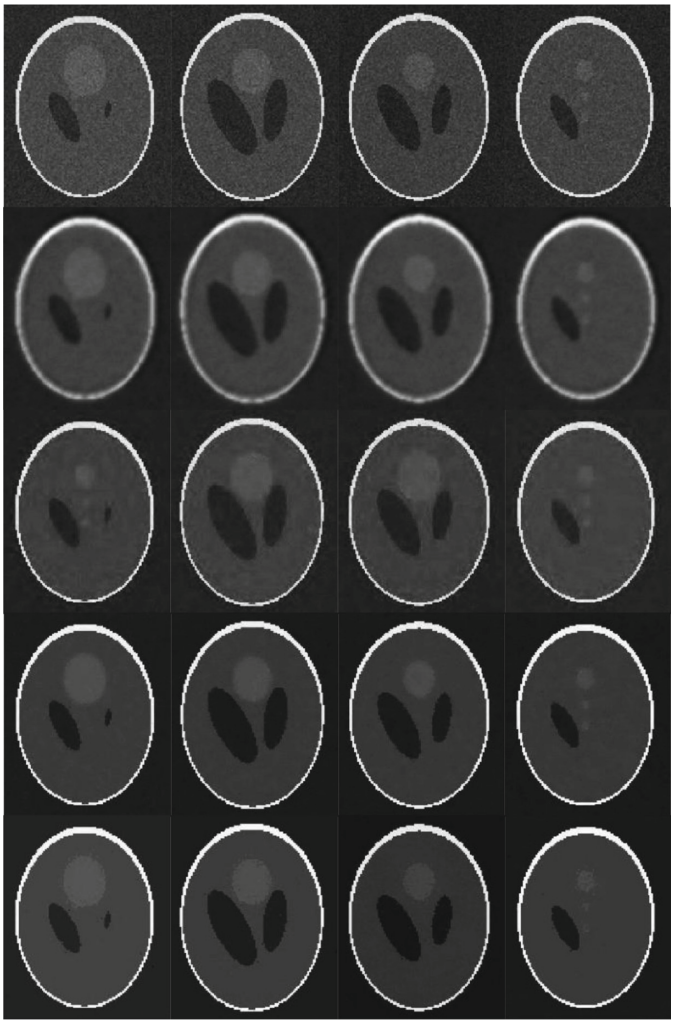


Fig. 3. Comparative results of denoising slices from 3D Shepp-Logan Phantom with 5% of add Gaussian White noise. From top to bottom are the results of: Noisy slices, Total Variation model (TV), Wavelet NeighShrinkSure denoised method, PFNLM method and our proposed method.

In addition, Tables 1 and 2 ensure these different results by giving the variable values obtained from the calculated quantitative measures. As shown in the tables, our model has the highest value of PSNR exceeds 40 dB in the different levels (3% and 5%) of noise which means that the denoising proceeding is of higher quality compared to the original image. Even the calculated SSIM value, which is too close to 1, means that our method have a good behavior against the different level of noise.

Table 1. Quantitative Measures for 3% Additive White Gaussian Noise.

Method	RMSE	PSNR	SSIM
Total variation	0.092	20.67	0.825
Wavelet NeighShrinkSure	0.015	36.32	0.942
PFNLM	0.005	45.90	0.973
Proposed method	**0.003**	**48.27**	**0.988**

Table 2. Quantitative Measures for 5% Additive White Gaussian Noise.

Method	RMSE	PSNR	SSIM
Total variation	0.092	20.69	0.775
Wavelet NeighShrinkSure	0.022	33.06	0.854
PFNLM	0.008	41.44	0.930
Proposed method	**0.007**	**43.66**	**0.986**

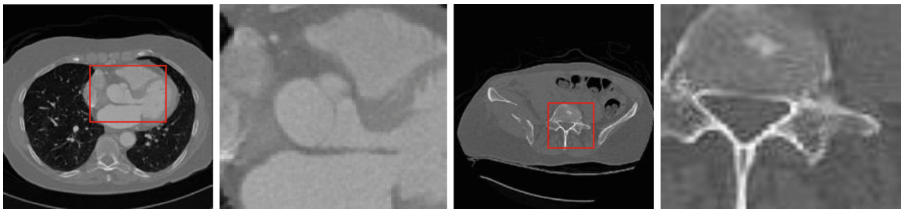


Fig. 4. Slice of real CT data. From the left to the right: transversal slices of a thoracic aorta, zoom details, transversal slices of a CT Pelvic, zoom detail.

As a real 3D CT images, we evaluate a volume ($512 \times 512 \times 347$) of a CT thoracic aorta of an adult and a volume ($512 \times 512 \times 225$) of a CT Pelvic images with fracture (Fig. 4) and from the online OSIRIX datasets [23]. We use visual assessment of the residual image in order to quantify the results since this data suffer from ground-truth.

Visually, all different denoised methods perform well against the noise in both the volumetric CT pelvis scan and CT thoracic aorta images, illustrated in the Figs. 5 and 6. Moreover, the residual images ensure the efficiently of our denoising method compared to the others algorithms. Our approach provides more accurate and remove noise softly with preserving edges and anatomical structures.

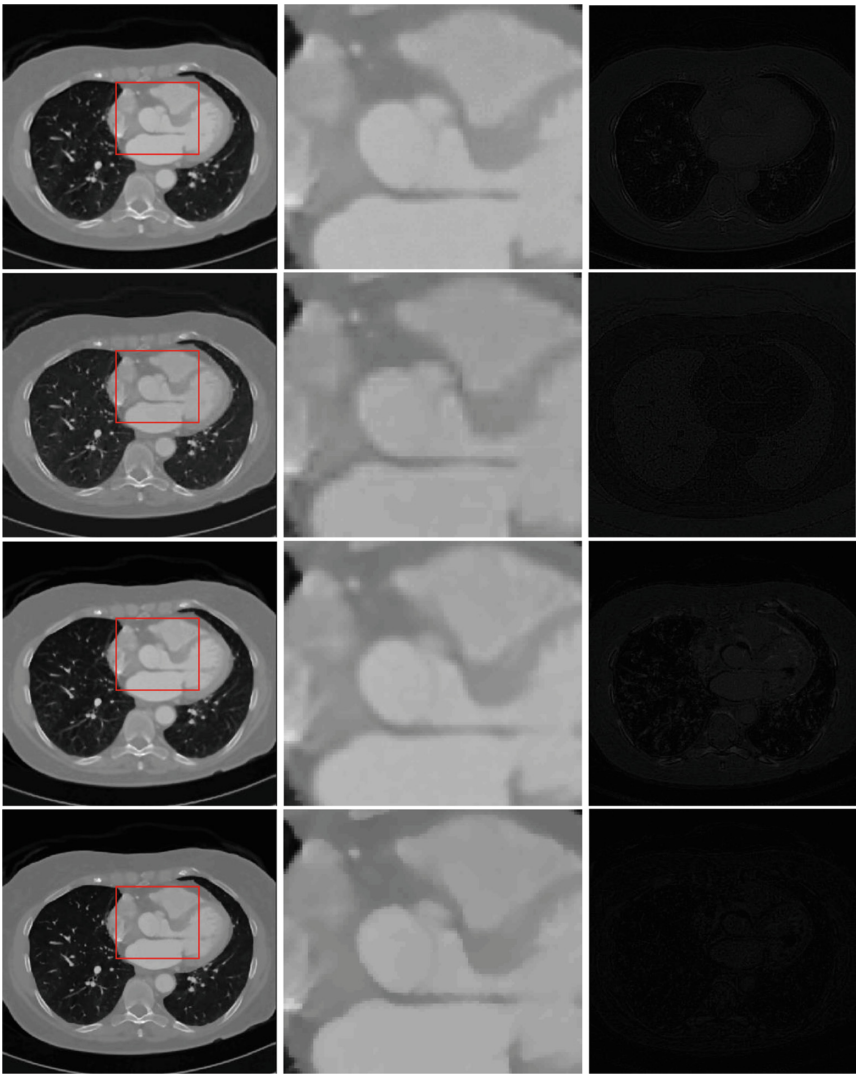


Fig. 5. Comparative results of denoising a CT thoracic aorta of an adult: From top to bottom: Total Variation model (TV), Wavelet NeighShrinkSure denoised method, PFNLM method and our proposed method. From the left to right: transversal slices, zoom details and the residual images.

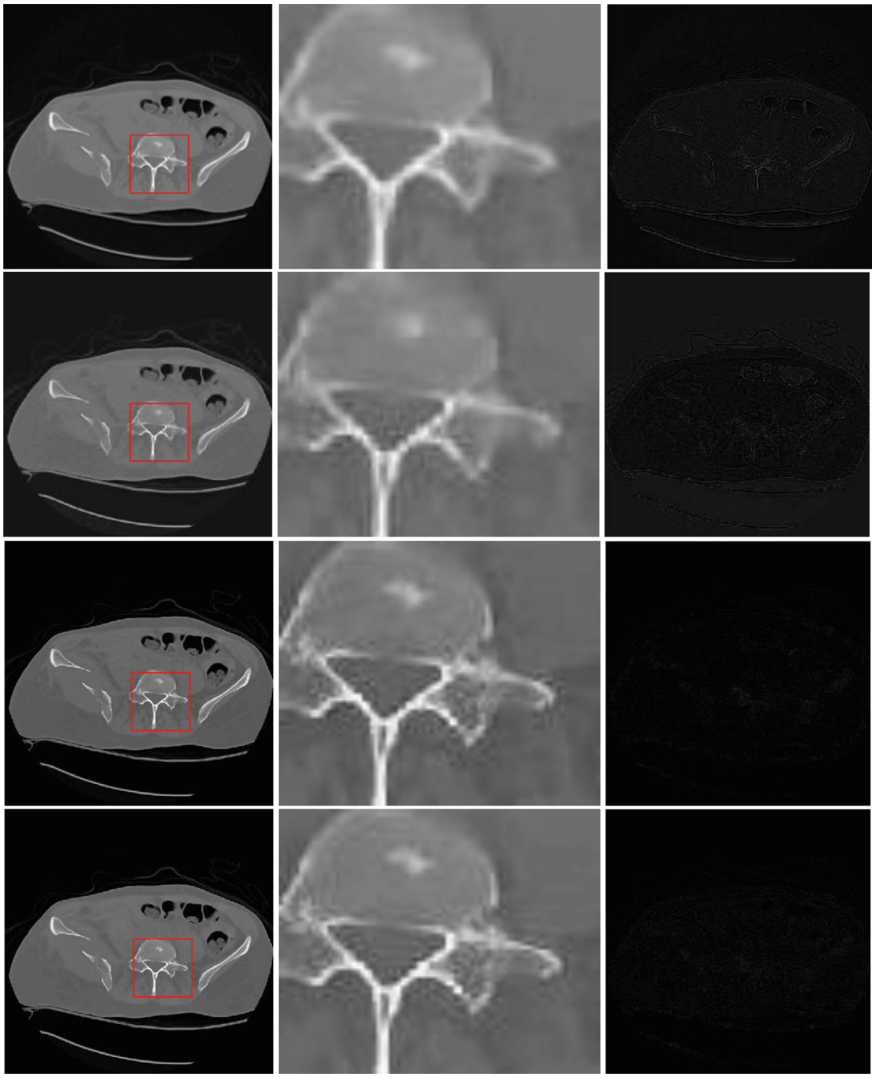


Fig. 6. Comparative results of denoising a CT Pelvic: From top to bottom: Total Variation model (TV), Wavelet NeighShrinkSure denoised method, PFNLM method and our proposed method. From the left to right: transversal slices, zoom details and the residual images.

5 Conclusion

A new hybrid method have been presented in this paper for denoising 3D CT data based on combination between two common filters such as the NL-mean filter and the diffusion tensor with MAD estimator for level noise in iteratively way. Different quantitative measures was computed in order to evaluate the efficiently of various denoised methods such as the RMSE, PSNR and SSIM index in a synthetic 3D CT data

which were promising in term of removing noise and preserving edges with highest measured values. In case of denoising real 3D CT data, our model outperforms the other methods by maintaining more significant anatomical structures. As a future work, we are looking to evaluate our proposed denoising model with segmentation task in order to extract the interesting regions.

References

1. Pratt, W.K.: Digital Image Processing. Wiley, New York (1978)
2. Lee, J.S.: Digital image enhancement and noise filtering by using local statistics. IEEE Trans. Pattern Anal. Mach. Intell. PAMI **2**, 165–168 (1980)
3. Chen, Z., King, R.: Breast volume denoising and noise characterization by 3D wavelet transform. ELSEVIER Comput. Med. Imaging Graph. **28**, 235–246 (2004)
4. Tran, M., Peteri, R., Bergounioux, M.: Denoising 3D medical images using a second order ariational model and wavelet shrinkage. Image Anal. Recogn. **7325**, 138–145 (2012)
5. Rudin, L.I., Osher, S., Fatemi, E.: Nonlinear total variation based noise removal algorithms. Phys. D. **60**, 259–268 (1992)
6. Chambolle, A., Lions, P.L.: Image recovery via total variation minimization and related problems Numerische Mathematik. J. Math. Imaging Vis. **77**, 167–188 (1997)
7. Bergounioux, M., Tran, M.P.: A second order model for 3D texture extraction. In: Bergounioux, M. (ed.) Mathematical Image Processing. Springer Proceedings in Mathematics. Springer, Berlin (2011)
8. Meijering, E., Niessen, W., Weickert, J., Viergever, M.: Diffusion-enhanced visualization and quantification of vascular anomalies in three-dimensional rotational angiography: results of an in-vitro evaluation. Med. Image Anal. **6**, 215–233 (2002)
9. Mendrik, A., Vonken, E., Rutten, A., Viergever, M., Ginneken, B.: Noise reduction in computed tomography scans using 3D anisotropic hybrid diffusion with continuous switch. IEEE Trans. Med. Imaging **28**, 1585–1594 (2009)
10. Frangakis, A., Hegerl, R.: Noise reduction in electron tomographic reconstruction using nonlinear anisotropic diffusion. J. Struct. Biol. **135**, 239–250 (2001)
11. Weickert, J., Haar, B.M, Viergever, M.A.: Conservative image transformations with restoration and scale-space properties. In: Proceedings of the International Conference on Image Processing, vol. 1, pp. 465–468 (1996)
12. Buades, A., Coll, B., Morel, J.M.: A review of image denoising algorithms, with a new one. Multiscale Model. Simul. **4**, 490–530 (2005)
13. Coupé, P., Yger, P., Prima, S., Hellier, P., Kervrann, C., Barillot, C.: An optimized blockwise NonLocal means denoising filter for 3-D magnetic resonance images. IEEE Trans. Med. Imaging **27**, 425–441 (2008)
14. He, L., Greenshields, I.R.: A nonlocal maximum likelihood estimation method for rician noise reduction in MR images. IEEE TMI **28**, 165–172 (2009)
15. Coupé, P., Manjon,V., Montserrat R., Collins, L.: Adaptive Multiresolution Non-Local Means Filter for 3D MR Image Denoising. IET Image Process. **6**(5), 558–568 (2012)
16. Romdhane, F., Benzarti, F., Amiri, H.: A new method for denoising three-dimensional magnetic resonance images denoising. Int. J. Comput. Vis. Rob. (in press)
17. Huber, P. J.: Robust statistics (1981)
18. Weickert, J.: Coherence-enhancing diffusion filtering. Int. J. Comput. Vis. **31**, 11–127 (1999)

19. Weickert, J.: Anisotropic diffusion in image processing (1998)
20. Wang, Y., Che, X., Ma, S.: Nonlinear filtering based on 3D wavelet transform for MRI denoising. EURASIP J. Adv. Sig. Process. **2012**, 1–14 (2012)
21. Vega, A.T., Pérez, V.G., Fernández, S.A., Westin, C.F.: Efficient and robust nonlocal means denoising of MR data based on salient features matching. Comput. Methods Programs Biomed. **105**, 131–144 (2012)
22. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: from error visibility to structural similarity. IEEE Trans. Image Process. **13**, 600–612 (2004)
23. http://www.osirix-viewer.com/datasets

Cost Sensitive Ranking Support Vector Machine for Multi-label Data Learning

Peng Cao[1(✉)], Xiaoli Liu[1], Dazhe Zhao[1], and Osmar Zaiane[2]

[1] Key Laboratory of Medical Image Computing of Ministry of Education, Computer Science and Engineering, Northeastern University, Boston, China
caopeng@cse.neu.edu.cn, liux3795@umn.edu, zhaodz@neusoft.com
[2] University of Alberta, Edmonton, Canada
zaiane@ualberta.ca

Abstract. Multi-label data classification has become an important and active research topic, where the classification algorithm is required to deal with prediction of sets of label indicators for instances simultaneously. Label powerset (LP) method reduces the multi-label classification problem to a single-label multi-class classification problem by treating each distinct combination of labels. However, the predictive performance of LP is challenged with imbalanced distribution among the labelsets, deteriorating the performance of traditional classifiers. In this paper, we study the problem of multi-label imbalanced data classification and propose a novel solution, called CSRankSVM (Cost sensitive Ranking Support Vector Machine), which assigns a different misclassification cost for each labelset to effectively tackle the problem of imbalance for Multi-label data. Empirical studies on popular benchmark datasets with various imbalance ratios of labelsets demonstrate that the proposed CSRankSVM approach can effectively boost classification performances in multi-label datasets.

Keywords: Multi-label learning · Imbalanced data · Classification · Rank SVM

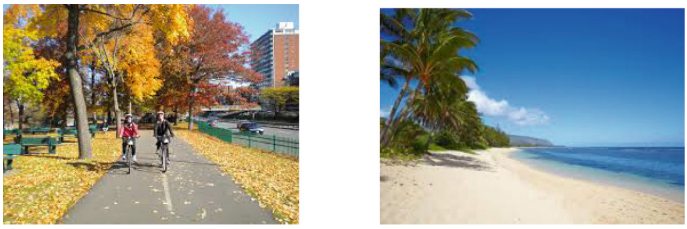
1 Introduction

In traditional label learning, each object is represented by a single instance and associated with a single label. Typically, binary classifiers are considered where only two classes exist, but in many applications more classes are used and we call these problems multi-class classification problems. Again each instance is associated with a single label. In multi-label classification problems, instances may be associated to more than just one label. Multi-label data classification has a wide variety of real world applications [1, 2], e.g. text categorization, scene classification, semantic video, annotation and biological data analysis.

Conventional multi-label learning algorithms aim to find a mapping from the feature space $X \subseteq \mathbb{R}^d$ to the label space $Y \subseteq \{0, 1\}^q$, wherein q is the number of labels. A simple yet effective multi-label learning method, called label powerset (LP) [3, 4], considers each distinct combination of labels that exist in the training set as a different

A. Abraham et al. (eds.), *Proceedings of the 16th International Conference on Hybrid Intelligent Systems (HIS 2016)*, Advances in Intelligent Systems and Computing 552,
DOI 10.1007/978-3-319-52941-7_25

class value of a single-label classification task, where each class denotes a unique 2^q-dimensional label vector. LP has the advantage of taking label correlations into consideration. However, the resulting distribution of the multiple labelsets (classes) is skewed, since many of these labelsets are usually associated with very few training examples due to a large number of labelsets appearing in the training set. Most multi-label data have hundreds of labels, with each instance being associated with a subset of them. Intuitively, it is easy to see that the more different labels exist, the more possibilities there are, and that some of them have a very low/high presence. For example, assuming an image database (samples in Fig. 1) and the task of scene classification, the labelset with the combination of *sunset* and *beach* could be more than the one of the *foliage* and *urban*. Also in textual repositories and document categorisation, documents have many labels and these are unevenly distributed. For the article classification, the labelset with the combination of *politics* and *economic* (e.g. Political risks may foil economic reform in China) could be more than the one of *sport* and *technology* (e.g. Professional sports teams are adopting advanced imagery technology to improve the performance of athletes and their recovery from injuries). The process of learning from imbalanced labelsets usually tends to be overwhelmed by the majority labelset and ignores the minority labelset examples, since most classifiers assume an even distribution of examples among classes and an equal misclassification cost. The imbalanced data issue has been deeply studied for single label classification [5]. This problem also affects multi-label datasets, and the imbalance level in multi-label datasets is much more significant than in binary or multi-class datasets in general.

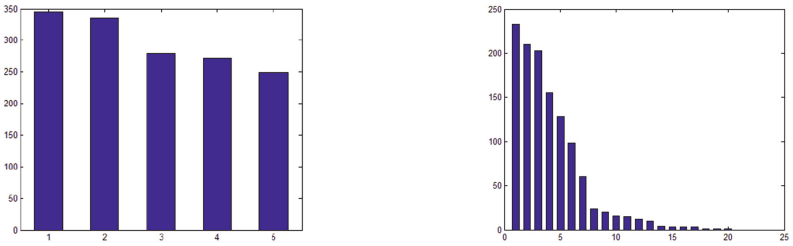


(a) **(b)**

Fig. 1. The example of imbalanced labelsets in multi-label learning (a) foliage and urban (b) sunset and beach

The issue of imbalanced data in multi-label learning can be tackled from two perspectives: individual label and labelset. In our work, we consider the class imbalance in the labelset space rather than the label space [6, 7] since the labelsets consider the high-order correlation among labels, exploiting the relation of multiple labels more effectively and intrinsically. Moreover, even when a dataset is balanced from the view of individual labels, which doesn't require imbalanced data learning, the imbalanced data issue still exists from the view of labelsets, such as the plant dataset (Fig. 2).

The common methods to solve data imbalance are data re-sampling perspective (i.e. undersampling large classes and oversampling small classes) and algorithm perspective (i.e. treating the imbalance problem in the algorithm itself). The main disadvantage of



(a) (b)

Fig. 2. The plant dataset (a) The histogram of individual label (b) The histogram of labelsets

re-sampling techniques is that they may cause loss of important information or introduce noisy data, since they change the original data distribution, especially for multi-label data with complicated label correlation, so as to resulting in model overfitting. To deal with the issue of imbalanced data in multi-label learning, we propose a novel labelset based multi-label classification method, called Cost sensitive Rank Support Vector Machines (CSRankSVM). The objective of CSRankSVM is to minimize the weighted ranking loss with the weighting scheme with respect to each sample while having a low complexity, so as to be able to learn more characteristics of samples with the minority labelset by setting a high cost to the misclassification of a minority labelset sample without modifying the data distribution.

Our proposed method combines both idea of problem transformation and the algorithm adaptation, both of which consider the label correlation when processing and learning. Firstly, the Label Powerset transforms the original data into multiple labelsets. This is problem transformation. Then, the CSRankSVM models the classifier by considering the imbalanced labelsets generated through assigning a different misclassification cost for relevant labelset of each instance. This is the algorithm adaptation. The contributions of this work can be listed as follows:

(1) First, we define a new metric to assess the imbalance level of labelsets generated by LP in the multi-label data, and propose two solutions to solve the imbalanced issue of labelsets, CSRankSVM and CSRankSVM-***p***.
(2) We empirically demonstrate that it improves the traditional RankSVM [9], and outperforms the state-of-the-art approaches for dealing with multi-label data on six benchmark datasets in terms of macro-FM, micro-FM and ranking loss.

2 Imbalanced Labelsets in Multi-label Data Learning

We formally define the multi-label classification problem as this: Let X denotes the space of instances and $Y = \{y_1, \ldots, y_q\}$ denotes the class labels where $|Y| = q$. $T = \{(x_1, Y_1), \ldots, (x_p, Y_p)\}$ $(|T| = p)$ is the multi-label training dataset. $Y_i \subset Y \subset 2^q$ is a labelset identity associated with instance $x_i \in X$, and the set Y' $(Yj, j = 1,\ldots,|Y|)$ is used

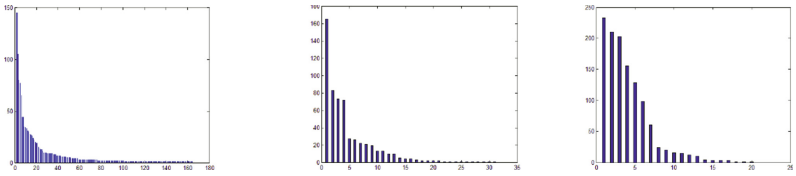
to denote the whole finite set labelset appeared in the training set. The goal of the multi-label classification is to get a classifier $h : X \rightarrow 2^q$ that generalizes well on both these training instances and unseen ones in the sense of optimizing some expected risk function with respect to a specific empirical loss function.

For binary imbalanced data, the class with fewer instances is the minority class, and the other class is the majority class. The imbalance level is easily measured taking into account only the two classes. However, for labelsets in multi-label data, there exist multiple majority labelsets (classes) and multiple minority labelsets (classes), therefore the imbalance is more challenging. Several interesting research questions are raised here: How to measure the imbalance level of labelsets for multi-label data? How to overcome the issue of imbalanced labelset learning in multi-label data? Is the imbalanced data learning method still effective for multi-label data? Can the classification performance of multi-label data be increased through tackling the imbalanced labelsets by the scheme of cost sensitive learning?

In traditional binary imbalanced data, the imbalanced data level between labels is assessed by the ratio between minority and majority examples. In this scenario of data with multi-labels, we need to introduce a new measure to assess the imbalance level of the whole dataset considering all the labels. We define the level of imbalance of labelsets, named *ImbalR*, which indicates a peakedness level of the histogram distribution of labelsets value in descending order, inspired by the idea of kurtosis.

$$ImbalR = \frac{\sum_{i=1}^{l} (Y_i - Y_{max})^4}{(l-1)s^4} \tag{1}$$

Where $s = \sqrt{\frac{1}{l}\sum_{i=1}^{l} (Y_i - Y_{max})}$, Y_i is the amount of instances in the *i-th* labelset, Y_{max} is the amount of instances of the labelset with the maximum amount of instances, l is the number of all the labelsets. If the histogram is more peaked (or inversely flatter), the distribution of labelsets value is more (or less) imbalanced, then *ImbalR* is larger (or smaller). The histogram of labelsets and *ImbalR* of multi-label datasets is shown in Fig. 3.



(a) (b) (c)

Fig. 3. The *ImbalR* of multi-label datasets (a) Yeast: *ImbalR = 1903* (b) Plant: *ImbalR = 381* (c) Image: *ImbalR = 43.4*

3 Cost Sensitive RankSVM

Current cost-sensitive learning research has been focused on binary or multi-class classification, but never yet on multi-label classification. Rank Support Vector Machine (RankSVM) [8] is an excellent kernel-based tool for multi label classification. However, the conventional SVM based method performs poorly on imbalanced learning because it pays less attention to the minority class [9, 10]. In order to overcome the imbalanced labelsets of multi-label classification, we inject the idea of cost sensitive learning into the RankSVM algorithm by coupling a cost λ_i to each instance. Therefore, we define the quadratic optimization problem of cost sensitive RankSVM as follows:

$$\begin{aligned} &\min \frac{1}{2}\sum_{k=1}^{q}\|w_k\|^2 + C\sum_{i=1}^{p}\frac{1}{|Y_i||\bar{Y}_i|}\sum_{(m,n)\in Y_i\times\bar{Y}_i}\lambda_i\xi_{imn}, \\ &\text{s.t.} <w_m - w_n, x_i> + b_m - b_n \geq 1 - \xi_{imn}, \quad \xi_{imn} \geq 0, (m,n) \in (Y_i \times \bar{Y}), i = 1, \ldots p. \end{aligned} \tag{2}$$

Dealing with imbalanced labelsets, cost-sensitive learning scheme assumes higher misclassification costs with minority labelset. The misclassification costs play a crucial role in the construction of a cost sensitive learning model for achieving expected classification results [10]. The most straightforward solution is to automatically generate the misclassification cost vector in accordance with the labelset distribution, which usually is in the form of a weight scheme inversely proportional to the number of samples in the labelsets. The relevant and minority labelsets of each instance are associated with higher misclassification cost values. Therefore, it is necessary to propose a RankSVM with cost sensitive learning to resolve the multi-label datasets.

The Lagrangian for the primal form in (2) can be expressed as (dual variables α_{imn} and η_{imn} related to the constraints of (2)):

$$\begin{aligned} L = &\frac{1}{2}\sum_{k=1}^{q}\|w_k\|^2 + C\sum_{i=1}^{p}\frac{1}{|Y_i||\bar{Y}_i|}\sum_{(m,n)\in Y_i\times\bar{Y}_i}\lambda_i\xi_{imn} \\ &- \sum_{i=1}^{p}\sum_{(m,n)\in Y_i\times\bar{Y}_i}\alpha_{imn}(<w_m - w_n, x_i> + b_m - b_n - 1 + \xi_{imn}) - \sum_{i=1}^{p}\sum_{(m,n)\in Y_i\times\bar{Y}_i}\eta_{imn}\xi_{imn} \end{aligned} \tag{3}$$

After some resolving according to the Karush-Kuhn-Tucker (KKT) conditions, we can get the following equations:

$$\partial_{w_k} L = w_k - \sum_{i=1}^{p}\left(\sum_{(m,n)\in Y_i\times\bar{Y}_i} c_{imn}\alpha_{imn}\right)x_i = 0; \tag{4}$$

$$\partial_{b_k} L = \sum_{i=1}^{p}\sum_{(m,n)\in Y_i\times\bar{Y}_i} c_{imn}\alpha_{imn} = 0;\ \partial_{\xi_{imn}} L = \frac{C\lambda_i}{|Y_i||\bar{Y}_i|} - \alpha_{imn} - \eta_{imn} = 0 \tag{5}$$

Where

$$c_{imn} = \begin{cases} 0 & \text{if } m \neq k \text{ and } n \neq k \\ +1 & \text{if } m = k \\ -1 & \text{if } n = k \end{cases} \tag{6}$$

By introducing (4)–(6) into the Lagrangian (3), the dual of the optimization problem can be expressed as:

$$\begin{aligned} &\max_{\alpha_{imn}} W(\alpha) = -\frac{1}{2}\sum_{k=1}^{q}\sum_{i,j=1}^{p}\beta_{ki}\beta_{kj}<x_i, x_j> + \sum_{i=1}^{p}\sum_{(m,n)\in Y_i\times\bar{Y}_i}\alpha_{imn}, \\ &\text{s.t.}\sum_{i=1}^{p}\sum_{(m,n)\in Y_i\times\bar{Y}_i} c_{imn}\alpha_{imn} = 0, \alpha_{imn} \in [0, \frac{C\lambda_i}{|Y_i||\bar{Y}_i|}], (m,n) \in (Y_i \times \bar{Y}), i = 1, \ldots p. \end{aligned} \tag{7}$$

where $\beta_{ki} = \sum_{(m,n)\in Y_i\times\bar{Y}_i} c_{imn}\alpha_{imn}$, $k = 1, \ldots q$

Then the Franke and Wolfe's method [8] is applied to solve the optimization problem, as done in the traditional RankSVM. Additionally, in order to reduce the excessive labelsets produced by LP, a Pruned Problem Transformation method (PPT) [11] is applied to prune the labelsets with a threshold (line 3—6), and the final pruned labelsets, labelsets-***p*** are obtained. The detailed algorithm CRankSVM (CRankSVM-***p***) is shown in Algorithm 1.

Algorithm 1. CSRankSVM (CSRankSVM-***p***)

Input: Training set D; Test set T; threshold parameter of pruning t;

1. Transform the original dataset D by LP into labelsets L
2. Compute the amount of instance in each labelset
3. **if** pruning the labelsets is required /* case for CSRankSVM-***p*** */
   for j=1 to $|L|$
4. Split the L_j into multi disjoint subsets where subsets occur more than t times in the D
5. The instances in L_j is duplicated and assigned one of the subsets
6. The original labelset L_j is discarded
   end for
   endif
   for i=1 to $|D|$
7. Obtain the relevant LabelSet $L(x_i)$ and irrelevant LabelSet $\overline{L(x_i)} = L \setminus L(x_i)$
8. Calculate the weight λ_i of x_i, λ_i =tansig($|L(x_i)|/|\overline{L(x_i)}|$)
   end for
9. Optimize CSRankSVM with weight vector λ according to Equations 3-7

4 Experimental Study

4.1 Experimental Setting

To evaluate the performance of our proposed method in multi-label classification tasks, a total of six common multi-label datasets are used in this study. Table 1 shows some useful statistics of these datasets, such as the number of instances in the training and test sets, the number of features (*Feat.*), the number of labels, label cardinality (*Card.*) and label density (*Dens.*). Moreover, we calculated the new measure of imbalanced data level of individual label, labelsets and labelsets-$\boldsymbol{p}$ for each datasets. In detail, the imbalance level is raised by the LP transformation except for *Scene*, and the distribution of labelsets becomes less skewed by pruning. As shown in Table 1, the imbalance labelset ratio (*ImbalR*) of the labelsets can be as low as 16.6 (13.0 for labelsets-$\boldsymbol{p}$), and the highest imbalance ratio happens to be 1903.3 (543.7 for labelsets-$\boldsymbol{p}$). In this work, for each RankSVM method, the RBF kernel is chosen (the kernel parameter is set to the mean value of distances between each pair of two instances), and the regularization parameter C is optimized by nested cross validation. We use macro-FM, micro-FM and ranking loss as evaluation criteria [6]. All the experiments are conducted by 10-fold cross-validation. The experimentation to our proposed CSRankSVM involves two stages:

Table 1. Statistics for six benchmark datasets used in our experiments

Datasets	Instances		Statistics				*ImbalR*		
	Training	Test	*Feat.*	*Labels*	*Card.*	*Dens.*	Label	Labelsets	Labelsets-*p*
Scene	1211	1196	294	6	1.074	0.179	24.3	16.6	13.0
Image	1200	800	294	5	1.236	0.247	10.8	43.4	34.4
Emotions	391	202	72	6	1.868	0.311	25.7	69.3	48.2
Plant	558	390	440	12	1.079	0.090	55.2	381.1	132.4
Human	1864	1244	440	14	1.185	0.085	66.5	1278.1	384.0
Yeast	1500	917	198	14	4.237	0.303	36.7	1903.3	543.7

Experiment I: To exhibit the influences of imbalanced labelsets and the performance of our proposed approaches, the comparison is conducted between our two methods and the traditional RankSVM in order to validate the effectiveness of cost sensitive learning experimentally;

Experiment II: We investigate the performance of CSRankSVM compared to six state-of-the-art methods. The results can confirm the advantages of our approach for multi-label data learning.

4.2 Experiment I

In this experiment, the comparison is conducted between our two methods and the basic RankSVM as well as **OS**RankSVM (RankSVM combined with random Over-Sampling) and **US**RankSVM (RankSVM combined with random Under-Sampling) [6] in terms of

Table 2. The performance of the three RankSVM methods

Methods	Metric	Yeast	Human	Plant	Scene	Emotions	Image
RankSVM	Macro-FM	0.3466	0.1637	0.1092	0.7084	0.6412	0.6358
	Micro-FM	0.6494	0.4306	0.3464	0.6981	0.6562	0.6310
	Ranking Loss	0.1743	0.1370	0.1801	0.0745	0.1716	0.1508
OSRankSVM	Macro-FM	0.3669	0.1744	0.1680	0.7166	0.6494	0.6267
	Micro-FM	0.6579	0.4455	0.3704	0.7084	0.6615	0.6226
	Ranking Loss	0.1728	0.1353	0.1725	0.0738	0.1749	0.1532
USRankSVM	Macro-FM	0.3369	0.1397	0.096	0.7137	0.6109	0.6176
	Micro-FM	0.6465	0.4095	0.3172	0.7046	0.6434	0.6094
	Ranking Loss	0.1779	0.1471	0.1896	0.0735	0.1741	0.1582
CSRankSVM	Macro-FM	0.3930	0.2414	0.2468	0.7130	0.6719	0.6432
	Micro-FM	0.6626	0.4357	0.3595	0.7051	0.6735	0.6387
	Ranking Loss	0.1653	0.1372	0.1656	0.0704	0.1525	0.1398
CSRankSVM-p	Macro-FM	0.3832	0.2334	0.2448	0.7170	0.6545	0.6654
	Micro-FM	0.6560	0.4330	0.3544	0.7094	0.6598	0.6591
	Ranking Loss	0.1661	0.1360	0.1694	0.0683	0.1533	0.1351

Macro-FM, Micro-FM, Ranking Loss and size of labelsets. Both re-sampling methods **OS**RankSVM and **US**RankSVM are among the most used preprocessing methods to equilibrate imbalanced datasets, and work in the labelset space as well. Table 2 summarizes the performance of the compared algorithms.

The results shows that the phenomenon of imbalanced labelsets affects the multi-label datasets, and our methods with a cost sensitive learning strategy can improve the performance for different imbalance degrees. That is to say, the separating boundary is moved towards the majority class so that additional minority samples are correctly classified, but slightly more majority samples are misclassified. Moreover, although CSRankSVM-***p*** does not improve the performance of CSRankSVM on the most datasets, the preprocessing of pruning can reduce the number of labelsets, obtains an improvement in efficiency but with inevitably slight information loss.

What is most important is that even the nearly balanced dataset from the label level, such as *Scene* and *Image*, benefited from the proposed methods. Therefore, from the view of individual label, the proposed weighted CSRankSVM is applicable to not only the imbalanced multi-label datasets, but also the well balanced datasets so long as the imbalanced labelsets exist.

4.3 Experiment II

In this section, we experimentally compare our CSRankSVM with five existing multi-label classification approaches involving data transformation strategy and algorithm adaptation strategy. The competing methods are BP-MLL (Backpropagation for Multi-Label Learning) [12], BR (binary relevance) [13], HOMER(Hierarchy Of Mul-

tilabel classifiERs) [14], CC(Classifier Chain) [15], RakEL(RAndom *k* labELsets) [3], CLR (Calibrated label ranking) [16]. The results are given in Tables 3, 4 and 5, in which the best performance for each dataset is highlighted. The numbers in parentheses denote the rank of the algorithm among the compared algorithms.

Table 3. The comparison of the proposed methods with respect to Macro-FM

Methods	Yeast	Human	Plant	Scene	Emotions	Image	AvgRank
BP-MLL	0.3392 (8)	0.0013 (8)	0.0219 (8)	0.0537 (8)	0.6379 (3)	0.2917 (8)	7.17
BR	0.3920 (5)	0.1527 (3)	0.1387 (5)	0.6285 (5)	0.5891 (6)	0.5414 (6)	5.00
HOMER	**0.4066 (1)**	0.1430 (5)	0.1549 (4)	0.6027 (7)	0.5601 (8)	0.5348 (7)	5.33
CC	0.3966 (3)	0.1498 (4)	0.1121 (6)	0.6126 (6)	0.5732 (7)	0.5437 (5)	5.17
RakEL	0.4031 (2)	0.1304 (7)	0.0765 (7)	0.7070 (3)	0.6313 (4)	0.6355 (3)	4.33
CLR	0.3834 (6)	0.1314 (6)	0.1552 (3)	0.6442 (4)	0.6242 (5)	0.5568 (4)	4.67
CSRankSVM	0.3930 (4)	**0.2414 (1)**	**0.2468 (1)**	0.7130 (2)	**0.6719 (1)**	0.6432 (2)	**1.83**
CSRankSVM-p	0.3832 (7)	0.2334 (2)	0.2448 (2)	**0.7170 (1)**	0.6545 (2)	**0.6654 (1)**	2.50

Table 4. The comparison of the proposed methods with respect to Micro-FM

Methods	Yeast	Human	Plant	Scene	Emotions	Image	AvgRank
BP-MLL	0.6452 (3)	0.0027 (8)	0.0234 (8)	0.1670 (8)	0.6608 (2)	0.3583 (8)	6.17
BR	0.5857 (7)	0.2949 (5)	0.2376 (4)	0.6194 (5)	0.6055 (6)	0.5395 (6)	5.50
HOMER	0.5858 (6)	0.2725 (7)	0.2362 (5)	0.5927 (7)	0.5748 (8)	0.5337 (7)	6.67
CC	0.5499 (8)	0.2960 (4)	0.2078 (6)	0.6001 (6)	0.5868 (7)	0.5419 (5)	5.83
RakEL	0.6254 (4)	0.2905 (6)	0.1267 (7)	0.6977 (3)	0.6467 (4)	0.6318 (3)	4.50
CLR	0.6158 (5)	0.3023 (3)	0.2564 (3)	0.6276 (4)	0.6364 (5)	0.5545 (4)	4.00
CSRankSVM	**0.6626 (1)**	**0.4357 (1)**	**0.3595 (1)**	0.7051 (2)	**0.6735 (1)**	0.6387 (2)	**1.33**
CSRankSVM-p	0.6560 (2)	0.4330 (2)	0.3544 (2)	**0.7094 (1)**	0.6598 (3)	**0.6591 (1)**	1.83

Table 5. The comparison of the proposed methods with respect to Ranking Loss

Methods	Yeast	Human	Plant	Scene	Emotions	Image	AvgRank
BP-MLL	0.1748 (3)	0.3934 (5)	0.4876 (8)	0.3986 (8)	0.1825 (4)	0.3430 (8)	6.00
BR	0.3097 (6)	0.4163 (8)	0.4772 (6)	0.2465 (6)	0.2941 (6)	0.2968 (5)	6.17
HOMER	0.3287 (8)	0.3989 (7)	0.4362 (5)	0.2345 (5)	0.3091 (8)	0.3055 (7)	6.67
CC	0.3238 (7)	0.3935 (6)	0.4861 (7)	0.2489 (7)	0.3028 (7)	0.2995 (6)	6.67
RakEL	0.2135 (5)	0.2311 (4)	0.1961 (3)	0.0998 (3)	0.1872 (5)	0.1807 (3)	3.83
CLR	0.1783 (4)	0.1571 (3)	0.2085 (4)	0.1011 (4)	0.1699 (3)	0.1917 (4)	3.67
CSRankSVM	**0.1653 (1)**	0.1372 (2)	0.1656 (2)	0.0704 (2)	**0.1525 (1)**	0.1398 (2)	1.67
CSRankSVM-p	0.1661 (2)	**0.1360 (1)**	**0.1694 (1)**	**0.0683 (1)**	0.1533 (2)	**0.1351 (1)**	**1.33**

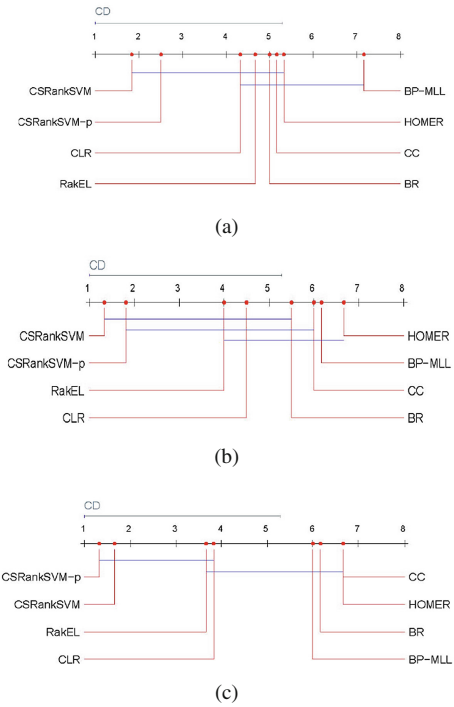


Fig. 4. Significant test of comparable methods (a) Significant test of Macro-FM, (b) Significant test of Micro-FM, (c)Significant test of Ranking Loss

To better understand the results of our techniques when compared to the other classification approaches, we performed a statistical analysis of our results. Firstly, a non-parametric Friedman test is used to determine that there is a statistically significant difference between the rankings of the classifiers in terms of G-mean and AUC. Consequently, we reject the null-hypothesis stating that all algorithms perform equally in mean ranking. Based on this rejection, the Nemenyi post-hoc test is used to compare all classifier to each other. It can be found from Fig. 4 that our methods statistically outperform BP-MLL in terms of Macro-FM, statistically outperform BP-MLL and HOMER in terms of Micro-FM, and statistically outperform CC, BR, BP-MLL and HOMER in terms of ranking loss.

5 Conclusion

This paper studies the challenges posed by the labelset imbalance problem. We introduce a new metric aimed to measure the imbalance level in multi-label datasets, and two cost sensitive methods designed to reduce the imbalance level of labelsets multi-label datasets are proposed. The experimental results on some benchmark multi-label data have demonstrated that the proposed methods provide a very competitive solution to other existing state-of-the-arts multi-label data classification methods.

Acknowledgments. This research was supported by the National Natural Science Foundation of China (61502091), the Fundamental Research Funds for the Central Universities (N140403004), and the Postdoctoral Science Foundation of China (2015M570254).

References

1. Zhang, M.L., Zhou, Z.H.: A review on multi-label learning algorithms. IEEE Trans. Knowl. Data Eng. **26**(8), 1819–1837 (2014)
2. Tsoumakas, G., Katakis, I.: Multi-label classification: an overview. Int. J. Data Warehouse. Min. **3**(3), 1–13 (2007)
3. Tsoumakas, G., Katakis, I., Vlahavas, I.: Random k-labelsets for multilabel classification. IEEE Trans. Knowl. Data Eng. **23**(7), 1079–1089 (2011)
4. Lo, H.Y., Lin, S.D., Wang, H.M.: Generalized k-labelsets ensemble for multi-label and cost-sensitive classification. IEEE Trans. Knowl. Data Eng. **26**(7), 1679–1691 (2014)
5. He, H., Garcia, E.A.: Learning from imbalanced data. IEEE Trans. Knowl. Data Eng. **21**(9), 1263–1284 (2009)
6. Charte, F., Rivera, A.J., del Jesus, M.J., Herrera, F.: Addressing imbalance in multilabel classification: measures and random resampling algorithms. Neurocomputing **163**, 3–16 (2015)
7. Tahir, M.A., Kittler, J., Yan, F.: Inverse random under sampling for class imbalance problem and its application to multi-label classification. Pattern Recogn. **45**, 3738–3750 (2012)
8. Elisseeff, A., Weston, J.: Kernel Methods for Multi-labelled Classification and Categorical Regression Problems. Technical report, BIOwulf Technologies (2001). http://www.kyb.tuebingen.mpg.de/bs/people/weston/publications

9. Bao, L., Juan, C., Li, J., Zhang, Y.: Boosted Near-miss Under-sampling on SVM ensembles for concept detection in large-scale imbalanced datasets. Neurocomputing **172**, 198–206 (2016)
10. Cao, P., Zhao, D., Zaiane, O.: An optimized cost-sensitive SVM for imbalanced data learning. In: Pei, J., Tseng, V.S., Cao, L., Motoda, H., Xu, G. (eds.) PAKDD 2013. LNCS (LNAI), vol. 7819, pp. 280–292. Springer, Heidelberg (2013). doi:10.1007/978-3-642-37456-2_24
11. Read, J.: A pruned problem transformation method for multi-label classification. In: Proceedings of the 2008 New Zealand Computer Science Research Student Conference (NZCSRS 2008), vol. 143150 (2008)
12. Zhang, M.L., Zhou, Z.H.: Multi-label neural networks with applications to functional genomics and text categorization. IEEE Trans. Knowl. Data Eng. **18**(10), 1338–1351 (2006)
13. Boutell, M.R., Luo, J., Shen, X., Brown, C.M.: Learning multi-label scene classification. Pattern Recogn. **37**(9), 1757–1771 (2004)
14. Tsoumakas, G., Katakis, I., Vlahavas, I.: Effective and efficient multilabel classification in domains with large number of labels . In: Proceedings of ECML/PKDD Workshop on Mining Multidimensional Data (MMD 2008), pp. 30–44 (2008)
15. Reed, J., Pfahringer, B., Holmes, G.: Classifier chain for multi-label classification. Mach. Learn. **85**(3), 333–359 (2011)
16. Fürnkranz, J., Hüllermeier, E., Mencía, E.L., Brinker, K.: Multilabel classification via calibrated label ranking. Mach. Learn. **73**(2), 133–153 (2008)

Sparse Learning and Hybrid Probabilistic Oversampling for Alzheimer's Disease Diagnosis

Peng Cao[1(✉)], Xiaoli Liu[1], Dazhe Zhao[1], and Osmar Zaiane[2]

[1] Key Laboratory of Medical Image Computing of Ministry of Education, College of Computer Science and Engineering, Northeastern University, Shenyang, China
caopeng@cse.neu.edu.cn

[2] Computing Science, University of Alberta, Edmonton, Canada

Abstract. Alzheimers Disease (AD) is the most common neurodegenerative disorder associated with aging. Early diagnosis of AD is key to the development, assessment, and monitoring of new treatments for AD. Machine learning approaches are increasingly being applied on the diagnosis of AD from structural MRI. However, the high feature-dimension and imbalanced data learning problem is two major challenges in the study of computer aided AD diagnosis. To circumvent this problem, we propose a novel formulation with hinge loss and sparse group lasso to select the discriminative features since features exhibit certain intrinsic group structures, then we propose a hybrid probabilistic oversampling to alleviate the class imbalanced distribution. Extensive experiments were conducted to compare this method against the baseline and the state-of-the-art methods, and the results illustrated that this proposed method is more effective for diagnosis of AD compared to commonly used techniques.

Keywords: Alzheimer's disease · Group lasso · Classification · Imbalanced data

1 Introduction

Alzheimers disease (AD) is the most frequent form of dementia in elderly patients, which causes progressive impairment of memory and other cognitive functions, leading directly to death. It accounts for 60–70% of age related dementia, affecting an estimated 30 million individuals in 2011 and the number is projected to be over 114 million by 2050 [1]. Early diagnosis of AD patients is important because it allows early treatment to improve the quality of life of the patients and their families. Therefore, effective and accurate diagnosis of AD, as well as its prodromal stage (mild cognitive impairment (MCI)), has attracted

P. Cao–Supported in part by National Natural Science Foundation of China (61502091).

A. Abraham et al. (eds.), *Proceedings of the 16th International Conference on Hybrid Intelligent Systems (HIS 2016)*, Advances in Intelligent Systems and Computing 552,
DOI 10.1007/978-3-319-52941-7_26

more and more attention recently. The machine learning, or classification approach has been used to provide markers for various neurological disorders including Alzheimer's disease [2,3] on structural magnetic resonance imaging (MRI). However, two of the key challenges in designing good prediction models for diagnosis of AD with MRI lie in:

(1) How to solve the dimensionality reduction in the high-dimensional data: High dimensionality of the data may affect the computational performance (processing time) and, worse, it may lead to a wrong estimation and identification of the relevant predictors [2]. Feature selection methods select a small subset of the original feature set to reduce the dimensionality of the data set and facilitate better generalization of training samples. In AD, features exhibit certain intrinsic group structures in the context of Alzheimer's disease diagnosis. There is natural grouping? of the features, the groups correspond to specific region-of-interest (ROIs) in the brain, and the individual features (average cortical thickness, standard deviation of thickness, surface area, cortical volume and subcortical volume) are specific properties of those regions. Hence the group effect in the features need to be taken into account while doing feature selection.
(2) How to address the imbalanced data distribution between different classes: In Alzheimer's Disease Neuroimaging Initiative (ADNI) dataset, the MCI cases eligible for the study are nearly two times the AD patients and normal controls (NC) [4]. This is a typical class imbalance problem? [5,6]. Class imbalanced data has detrimental effects on the performance of conventional classifiers. The actual cause for the poor performance of conventional classifiers on the minority class is not necessarily related to only the between-class imbalance. Within-class imbalance [7] refers to the case where a class is formed of a number of sub-clusters with different sizes, concerns itself with the distribution of representative data for sub-concepts within a class. The existence of sub-concepts also increases the complexity of the problem because the amount of instances among them is not usually balanced. For the patients with AD or MCI, there exists multiple sub-concepts as the disease involves multiple different subtype or different characteristic, which results in the distribution of instances over each class concepts and may yield clusters with unequal sizes.

The aim of our work is to simultaneously address both issues, and to explore whether oversampling combined with feature selection benefits the diagnosis of individuals based on multivariate brain MRI data. A low-dimensional representation of the data not only reduces the risk of overfitting, improving the models generalization ability, but also allow the oversampling algorithm to generate much accuracy synthetic instances [15]. Based on the motivation, we propose a new formulation incorporating hinge loss combined with sparse group lasso (called HLSGL) to conduct the feature selection. The objective function is non-convex and non-smooth, which is difficult to solve in general. To address it, we use differentiable approximation of hinge loss, and adapt a accelerated proximal

gradient method for solving the non-smooth formulation [9]. On the other hand, we propose a hybrid probabilistic oversampling (HPS) algorithm to balance the skew class distribution based on the lower dimensional data. Experimental results show that HLSGL-HPS achieves a considerable improvement in the prediction performance. Results also demonstrate that feature selection and oversampling are two key steps for diagnosis of AD before building classifier model, and feature selection can be very helpful when facing imbalanced data sets.

2 Proposed Method

In this section, we describe the proposed approach in diagnosis of AD with structural MRI brain images. At first, feature selection based on HLSGL is performed to select relevant ROIs and features, then HPS is conducted to balance the skew class distribution on the selected feature subset. Finally, we build a kernel classifier trained on lower dimensional balanced data to classify AD/NC, AD/MCI and MCI/NC. HLSGL not only reduces the dimensionality so as to improve the performance of the HPS algorithm, but also makes the classification model interpretable. Moreover, HPS offers an effective solution for within-class in tandem with the between-class imbalance according to the distribution probability without jeopardizing structure of data.

2.1 Feature Selection by Hinge Loss Combined with Sparse Group Lasso, HLSGL

The general goal of supervised learning is to predict for the input $\boldsymbol{x}$ an output y. To achieve this objective, learning algorithms usually use training data $\{\boldsymbol{x}_i, y_i\}_{i=1}^n$ to learn a prediction function f that can correlate $\boldsymbol{x}$ with y. A common approach to obtain is to minimize the following regularized empirical error:

$$\min_{\boldsymbol{w}\in\mathbb{R}^P} l(\boldsymbol{w}; \boldsymbol{y}, \boldsymbol{X}) + \lambda r(\boldsymbol{w}) \tag{1}$$

where $l(\cdot)$ denotes the loss function, $r(\cdot)$ is the regularizer and λ is regularization parameter. In the context of classification, we employ hinge loss $l(\boldsymbol{w}) = max(1 - \boldsymbol{y}\boldsymbol{w}^T\boldsymbol{X}, 0)$, because kernel methods [12,13] have been shown to be very effective for classification in Alzheimers disease [10,11], and hinge loss is used in the objective function of kernel methods. However, hinge loss is non-smooth loss function, which is difficult to optimize. Therefore, we propose to use a differentiable approximation of hinge loss, which is defined as:

$$l(y,t) = \begin{cases} 0 \quad if \quad yt > 1+h \\ \dfrac{(1+h-yt)^2}{4h} \quad if \quad |1-yt| \le h \\ 1-yt \quad if \quad yt < 1-h \end{cases} \tag{2}$$

where $t = \boldsymbol{w}^T\boldsymbol{X}$ is the predicted label, and h is a parameter to choose, typically between 0.01 and 0.5.

Sparse methods have attracted a great amount of research efforts in the past decade due to its sparsity-inducing property and strong theoretical guarantees [14]. Traditionally the ℓ_1-norm (namely lasso) was effectively implemented to feature selection in high-dimensional setting. However, lasso fails to capture the correlation information among the pairwise of group features. For group of features that the pairwise correlations among them are very high, lasso tends to select only one of the pairwise correlated features and does not induce the group effect. In the context of AD, the groups correspond to specific regions of-interest (ROIs) in the brain, and the individual shape features are specific properties of those regions. The multiple shape measures from the same ROI tend to be selected together as joint predictors, and use this prior knowledge to group relevant shape features together in the same ROI guide the learning process. Let $G = \{G_1, \ldots, G_m\}$ be the groups of variables at the m ROIs considered. For our data, the number of features in each group is 4 for cortical region involving cortical thickness average (TA), standard deviation of thickness (TS), surface area (SA) cortical volume (CV), and 1 for subcortical region involving subcortical volume (SV).

The group lasso is a technique to do variable selection on (predefined) groups of variables [8,16]. However, it is a strict assumption for feature selection. Our motivation in promoting structured sparsity is drawn from the fact that for predicting disease status, if a particular brain region (ROI) is irrelevant, then coefficients of all morphological features at that ROI should be zero. Furthermore, if a particular brain region is deemed relevant, then we should be able to select the most important? morphological feature(s) at that region to be considered for prediction. In order to achieve the goal of structural sparse feature selection (SGL), we employ sparse group lasso integrating lasso $\|\boldsymbol{w}\|_1$ and group lasso $\|\boldsymbol{w}\|_{G_{2,1}}$ as feature selection with structural grouping sparsity, so as to allow simultaneous joint feature selection from feature level and ROI level. The hierarchical norm of SGL is defined as:

$$r(\boldsymbol{w}) = \lambda_1 \|\boldsymbol{w}\|_1 + \lambda_2 \|\boldsymbol{w}\|_{G_{2,1}} \tag{3}$$

where $\|\boldsymbol{w}\|_1 = \sum_{i=1}^{p} \|\boldsymbol{w}_i\|_2$, and $\|\boldsymbol{w}\|_{G_{2,1}} = \sum_{j=1}^{m} \sqrt{\sum_{i \in G_j} \left\|\boldsymbol{U}_{G_j}\right\|_2^2}$. Note that p and m is the size of features and ROIs size, respectively.

By approximating hinge loss, $l(\boldsymbol{w}; \boldsymbol{y}, \boldsymbol{x})$ is convex and differentiable with Lipschitz continuous gradient so that $\|f(\boldsymbol{z}) - f(\boldsymbol{w})\| \leq L\|\boldsymbol{z} - \boldsymbol{w}\|$ where L denotes the Lipschitz constant, however $r(\boldsymbol{w})$ is still convex but non-smooth. For the convex but non-smooth formulation, we solve it by designing a new accelerated proximal gradient (APG) method [9] in this work.

A well studied idea in efficient optimization of such composite objective function is to start with a quadratic approximation of the form: $Q_L(\boldsymbol{w}, \boldsymbol{w}^{(t)}) := l(\boldsymbol{w}^{(t)}) + \langle \boldsymbol{w} - \boldsymbol{w}^{(t)}, \nabla l(\boldsymbol{w}^{(t)}) \rangle + \frac{L}{2}\|\boldsymbol{w} - \boldsymbol{w}^{(t)}\|^2 + \lambda r(\boldsymbol{w})$. Ignoring constant terms in $\boldsymbol{w}^{(t)}$, the unique minimizer of the above expression can be written as $\pi_L^{l,r}(\boldsymbol{w}^{(t)}) = \arg\min_{\boldsymbol{w}} \left\{ r(\boldsymbol{w}) + \frac{L}{2}\|\boldsymbol{w} - \left(\boldsymbol{w}^{(t)} - \frac{1}{L}\nabla l(\boldsymbol{w}^{(t)})\right)\|^2 \right\}$, which can be viewed as a proximal operator corresponding to the non-smooth function $r(\boldsymbol{w})$.

A popular approach to solving the non-smooth problems is to simply do the following iterative update: $\boldsymbol{w}^{(t+1)} = \pi_L^{l,r}(\boldsymbol{w}^{(t)})$, which can be shown to have a $O(1/t)$ rate of convergence [19]. In practice, since the Lipschitz constant L may be unknown, we follow the adaptive strategy suggested in [9] to make sure we make progress.

For our purposes, we consider a refined version of the iterative algorithm inspired by Nesterov's accelerated gradient descent [19]. The main idea, as studied in the literature as APG algorithms [9], is to iteratively consider the proximal operator $\pi_L^{l,r}(\cdot)$ at a specific linear combination of the previous two iterates $\{\boldsymbol{w}^{(t)}, \boldsymbol{w}^{(t-1)}\}$, in particular at $\hat{\boldsymbol{w}}^{(t)} = \boldsymbol{w}^{(t)} + \alpha_{t+1}(\boldsymbol{w}^{(t)} - \boldsymbol{w}^{(t-1)})$ instead of at just the previous iterate w_t. The choice of α_{t+1} follows Nesterov's accelerated gradient descent [19] and is detailed in Algorithm 1. The iterative algorithm simply updates $\boldsymbol{w}^{(t+1)} = \pi_L^{l,r}(\hat{\boldsymbol{w}}^{(t)})$

As shown in [9], the algorithm has a rate of convergence of $O(1/t^2)$. With $\tilde{\boldsymbol{w}}^{(t+1)} = (\hat{\boldsymbol{w}}^{(t)} - \frac{1}{L}\nabla l(\hat{\boldsymbol{w}}^{(t)}))$, the problem of computing the proximal operator $\pi_L^{l,r}(\hat{\boldsymbol{w}}^{(t)}) := T_L^{\lambda_1,\lambda_2}(\tilde{\boldsymbol{w}}^{(t)})$ is given by $T_L^{\lambda_1,\lambda_2}(\tilde{\boldsymbol{w}}^{(t)}) = \arg\min_{\boldsymbol{w}\in\mathbb{R}^p}\{\lambda_1 \|\boldsymbol{w}\|_1 + \lambda_2 \|\boldsymbol{w}\|_{G_{2,1}} + \frac{L}{2}\|\boldsymbol{w} - \tilde{\boldsymbol{w}}^{(t)}\|^2\}$.

The proximal operator $T_L^{\lambda_1,\lambda_2}(\tilde{\boldsymbol{w}}^{(t)})$ can be computed efficiently in two steps, as outlined below:

$$\tilde{\boldsymbol{s}}^{(t)} = T_L^{\lambda_1}(\tilde{\boldsymbol{w}}^{(t)}) \ , \tag{4}$$

$$\boldsymbol{w}^{(t+1)} = T_L^{\lambda_2}(\tilde{\boldsymbol{s}}^{(t)}) = T_L^{\lambda_1,\lambda_2}(\tilde{\boldsymbol{w}}^{(t)}) \ . \tag{5}$$

$T_L^{\lambda_1}(\tilde{\boldsymbol{w}}^{(t)})$ can be obtained by soft-thresholding directly $\tilde{\boldsymbol{s}}^{(t)} = \mathrm{sgn}(\tilde{\boldsymbol{w}}^{(t)})\max\{|\tilde{\boldsymbol{w}}^{(t)}| - \lambda_1, 0\}$. Following [17], the group lasso updates can be done by:

$$\boldsymbol{w}_j^{(t+1)} = T_L^{\lambda_2}(\tilde{\boldsymbol{s}}^{(t)}) = \frac{\max\{\|\tilde{\boldsymbol{s}}_j^{(t)}\|_F - \frac{\lambda_2}{L}, 0\}}{\|\tilde{\boldsymbol{s}}_j^{(t)}\|_F}\tilde{\boldsymbol{s}}_j^{(t)} \ . \tag{6}$$

where $\boldsymbol{w}_j^{(t+1)}, \boldsymbol{s}_j^{(t+1)}$ are group specific sub-vector correspond to group j in $\boldsymbol{w}^{(t+1)}, \boldsymbol{s}^{(t+1)}$ respectively.

2.2 Hybrid Probabilistic Sampling

In order to solve the between-class and within-class imbalance simultaneously, we propose a hybrid probabilistic sampling (HPS) with the combination of over-sampling and under-sampling, and incorporates probability function in its data distribution re-sampling mechanism. It generates more accurate instances to generalize the decision region for the minority class, and removes the redundant instances for the majority class without destroying the structure of the data. It can deal with the between-class imbalance and within-class imbalance issues simultaneously.

Gaussian Mixture Models (GMM) are generative probabilistic models of several Gaussian distributions for density estimation in machine learning applications. A Gaussian mixture can be constructed to acceptably approximate any given density. Therefore, we assume the distribution of two classes follows the Gaussian mixture model with unknown parameters. The parametric probability density function of GMM is defined as a weighted sum of Gaussians. The finite Gaussian mixture model with k components may be written as: $p(y|\mu_1,\ldots,\mu_k;\sigma_1,\ldots,\sigma_k;\pi_1,\ldots,\pi_k) = \sum_{j=1}^{k}\pi_j N(\mu_j,\sigma_j)$, and $0 \leq \pi_j \leq 1$, $\sum_{j=1}^{k}\pi_j = 1$, where μ_j are the means, σ_j are covariance matrixes, π_j are the mixing proportions, and $N(\mu_j,\sigma_j)$ is a Gaussian with specified mean and variance.

We estimate the parameters of GMM with FJ algorithm [18], which can overcome the major weaknesses of the basic EM algorithm particularly vis-à-vis the initialization, and can automatically select the number of component, we use it here to estimate the parameters of GMM. Each instance $\boldsymbol{x}_i$ will then be assigned to the cluster k where it has the largest posterior probability $p(k|\boldsymbol{x}_i)$. When calculating the probability of each instance on each component, the probabilities for the feature is obtained by a Gaussian density function. At the same time, we obtain the parameters of each Gaussian component. For different clusters, the re-sampling rates are different; within the cluster, the probabilities of each instance to be chosen for re-sampled are different.

We use the oversampling combined with undersampling to balance the class size. The sizes of the two classes are N^+ and N^-. The gap N_G between two uneven classes is: $N_G = N^- - N^+$. Thus, the amount of instances in the minority class for oversampling is: $N_\alpha^+ = N_G \times \alpha$, and the amount in the majority class for undersampling is: $N_\alpha^- = N_G \times (1-\alpha)$. To adjust the within class imbalance, we need to balance cluster sizes in each class. For the minority class, the number of instances to be oversampled is inversely proportional to the size of the cluster; for the majority class, the numbers of instances to be undersampled are proportional to the size of the cluster. Furthermore, we use the probabilities of each instance to conduct the re-sampling with maintaining the data structure, in order to address the two type imbalance issues. In the clusters of the majority class, the instances with higher probability are dense, they are frequent in the subclass, and hence they have higher chance to be undersampled. We choose the instances to be undersampled according to the Gaussian distribution. In the clusters of the minority class, the new instances are produced according to the probability function of Gaussian distribution, resulting in finding more potentially interesting regions. The main steps in undersampling for the clusters of the majority class and oversampling for the clusters of the minority class are conducted as following:

Step 1: In the oversampling for the minority class, the smaller the size of cluster within the class, the more instances are over-sampled, so as to avoid the small disjuncts. For the *i-th* cluster, the amount of synthetic instances needed to be generated is: $N_\alpha^{i+} = (\frac{1}{size_+^i}/\sum_{j=1}^{S_+}\frac{1}{size_+^j}) \times N_\alpha^+$, where $size_+^i$ is the size of *i-th* cluster, S_+ is the number of clusters in the minority class.

Step 2: In the *i-th* cluster, N_{α}^{i+} instances are generated with the parameters from the current Gaussian distribution. The new instances are generated according to the probability function of the Gaussian distribution with parameters learned from the available data. Firstly, the probability from the Gaussian distribution of each instance is calculated and normalized: $\hat{p_k} = p_k / \sum_{j=1}^{size_+^i} p_j$. Then, the amount of new instances for each instance x_k is obtained according to: $n_k = N_{\alpha}^{i+} \times \hat{p_k}$. For ensuring that synthetic instances created via this method always lay in the region near $\boldsymbol{x}_k$, the n_k instances are generated in its $K(K = 5)$ nearest neighbors region. It can extend more potential regions rather than being limited along the line between the minority example and its selected nearest neighbors. In addition, this guarantees the creation of minority samples in the cluster, and avoids any incorrect synthetic instance generation.

Step 3: In the undersampling for the majority class, we calculate the amount of instances to be under-sampled for each cluster. The number of instances to be under-sampled are proportional to the size of clusters. For the *i-th* cluster, the amount of instances needed to be removed is: $N_{\alpha}^{i-} = (size_-^i / \sum_{j=1}^{S_-} size_-^j) \times N_{\alpha}^{-}$, where $size_-^i$ is the size of *i-th* cluster, S_- is the number of clusters in the majority class.

Step 4: In each component Gaussian distribution, the center region is denser than the border region. These instances from the center are more possible to be redundant, and so are better candidates to be under-sampled. We need to choose the instances to be ignored or removed located on the center of the distribution more than the border. The probabilities to be chosen for undersampling are proportional to the normalized probability $\hat{p}$ of the Gaussian distribution for each instance in a cluster.

3 Experimental Study

3.1 Dataset and Setting

The data used in this paper were obtained from the ADNI database (adni.loni.usc.edu) [21]. The MRI features used in our experiments are based on the imaging data from the ADNI database processed by a team from UCSF (University of California at San Francisco), who performed cortical reconstruction and volumetric segmentations with the FreeSurfer image analysis suite (http://surfer.nmr.mgh.harvard.edu/). There were $p = 319$ MRI features (covariates) in total. In this work, only ADNI1 subjects with no missing feature and cognitive outcome information baseline data are included. This yields a total of $n = 788$ subjects, who are categorized into 3 baseline diagnostic groups: Cognitively Normal (CN, $n_1 = 225$), Mild Cognitive Impairment (MCI, $n_2 = 390$), and Alzheimer's Disease (AD, $n_3 = 173$).

For the HLSGL, the regularization parameters λ_1, λ_2 in Eq. (1) and C in SVM are chosen by nested cross-validation strategy on the training data (trying values 0.01, 0.1, 1, 10, 100, 1000). In HPS, α is set to 70%. To evaluate the

performance of different classification methods, we use a 10-fold cross-validation strategy to compute the sensitivity, the specificity, G-mean and AUC.

3.2 Effectiveness of Proposed Kernel Framework

To evaluate and compare the performances of each method, three classification experiments were performed: AD vs CN, AD vs MCI and MCI vs NC. We compare our method (HLSGL-HPS) with five baseline methods involving SVM working on the original dataset (ALL), hinge loss combined lasso (HLL), HPS on the original dataset (All-HPS) and HLL combined with HPS (HLL-HPS). In the experiment of CN vs AD, the two class are nearly balanced, therefore oversampling is not need to be conducted. Prediction performance results of three groups, measured by five measure metrics of HLSGL-HPS and baseline methods are shown in Table 1. The results demonstrate that reducing the feature dimensionality can mitigate the overfitting problem and improve a models generalizability. Moreover, sparse group lasso based feature selection can achieve robust classification performance compared with lasso. Furthermore, the results indicate that feature selection is as important as the oversampling in the imbalanced data classification. Oversampling on the high dimensional irrelevant feature leads to the creation of wrong instances when the class dispersion or the class noise exists.

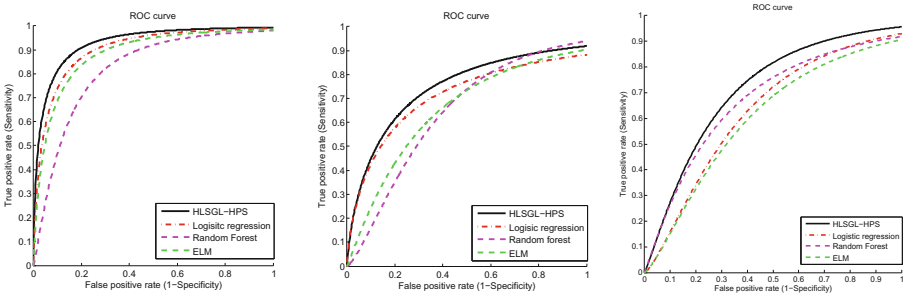
Moreover, we provides an empirical evaluation of the proposed method compared with other classification methods commonly used in diagnosis of AD, such as ELM (Extreme learning machine) [3], logistic regression [22] and random

Table 1. Comparison of baseline methods and HLSGL-HPS

	AD vs. NC				AD vs. MCI				MCI vs. NC			
	Sen	Spec	G-mean	AUC	Sen	Spec	G-mean	AUC	Sen	Spec	G-mean	AUC
All	0.802	0.932	0.865	0.859	0.277	0.905	0.501	0.510	0.342	0.865	0.544	0.433
HLL	0.811	0.941	0.874	0.879	0.253	**0.928**	0.485	0.523	0.208	**1**	0.456	0.517
HLSGL	**0.829**	**0.948**	**0.882**	**0.892**	0.334	0.919	0.536	0.571	0.253	0.965	0.494	0.594
All-HPS	-	-	-	-	0.586	0.737	0.657	0.682	0.418	0.687	0.536	0.593
HLL-HPS	-	-	-	-	0.645	0.672	0.659	0.671	0.544	0.692	0.613	0.606
HLSGL-HPS	-	-	-	-	**0.691**	0.696	**0.693**	**0.710**	**0.646**	0.576	**0.650**	**0.682**

Table 2. Average values of G-mean and AUC for all compared classification methods (Note that $\star$ stands for the case with $p \leq 0.05$)

Methods	G-mean			AUC		
	AD vs NC	AD vs MCI	MCI vs NC	AD vs NC	AD vs MCI	MCI vs NC
Random Forest	0.861*	0.609*	0.632	0.833*	0.707	0.653*
ELM	0.855*	0.661*	0.631	0.857*	0.691*	0.636*
Logistic regression	0.727*	0.652*	0.619*	0.820*	0.663*	0.617*
Proposed method	**0.882**	**0.693**	**0.650**	**0.892**	**0.722**	**0.682**



(a) AD vs. NC (b) MCI vs. NC (c) AD vs. MCI

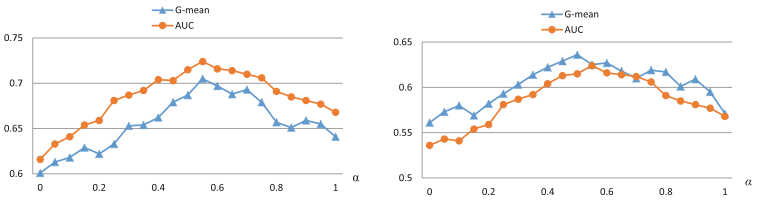
Fig. 1. The ROC of multiple computing methods

forest [23] with respect to G-mean and AUC in Table 2. From Table 2 and Fig. 1 we observe that our proposed framework is always better than the other three classification methods, even significantly outperform them in the most cases.

We applied HLSGL to conduct the ROI selection. In each fold, the ROIs are ranked based on the weight value corresponding to this ROI by frobenius norm, and the top-10 are identified as important. The important regions identified such as Hippocampus, Entorhinal and ParsOpercularis are relevant according to existing AD domain knowledge and in accordance with prior studies [2,20].

3.3 The Effect of Re-Sampling Ratio on the Performance

In HPS, the optimal re-sampling ratio α may be unknown, and the parameter plays a vital role for the performance of hybrid re-sampling on the imbalanced data learning. In the previous experiments, α is set to 70% empirically. The experiment shows the performance by tuning the re-sampling ratio. The range of α is [0, 100%]; the step is set to 5%. With each value of α, we conduct 10-fold cross validation to obtain an averaged G-mean and AUC results. From Fig. 2, we can see the changes of G-mean and AUC when varying the value of α. When α is 0, only undersampling for majority class is carried out and no new instances are generated. Important information of majority class may be lost, hence the performance is lowest. When increasing the value of α, the two performances increase. When α is 1, oversampling for minority class is performed without removing redundant instances for majority class. The issue of overfitting may occur due to the large amount of the minority class as well as the redundant information of majority class. Moreover, we found the G-mean and AUC to achieve the best when α is 55% for AD vs MCI, and 45% as well as 55% for MCI vs NC respectively. It demonstrates that the hybrid re-sampling scheme with an appropriate re-sampling ratio can achieve optimal classification performance.



(a) AD vs MCI (b) MCI vs NC

Fig. 2. The performance of varying oversampling ratio with respect to G-mean and AUC

4 Conclusion

In this paper, we focused on the high feature-dimension and class imbalanced distribution problem in AD diagnosis. Specifically, we proposed a novel feature selection method by sparse group lasso and hybrid probabilistic oversampling to preprocess the high dimensional imbalanced data, to improve the prediction performance. In our experimental results on the ADNI dataset, we validated the efficacy of the proposed method by enhancing classification performance in terms of G-mean and AUC. In our future works, we will extend the proposed linear sparse feature selection to the nonlinear sparse modal via kernel functions to capture complex patterns for AD diagnosis.

Acknowledgments. This research was supported by the National Natural Science Foundation of China (61502091), the Fundamental Research Funds for the Central Universities (N140403004), and the Postdoctoral Science Foundation of China (2015M570254).

References

1. Brookmeyer, R., Johnson, E., Ziegler-Graham, K.: Forecasting the global burden of Alzheimers disease. Alzheimer's Dement. **3**(3), 186–191 (2007)
2. Zhu, X., Suk, H., Shen, D.: Subspace regularized sparse multi-task learning for multi-class neurodegenerative disease identification. IEEE Trans. Biomed. Eng. **63**(3), 607–618 (2015)
3. Peng, X., Lin, P., Zhang, T., Wang, J.: Extreme learning machine-based classification of ADHD using brain structural MRI data. PloS One **8**(11), e79476 (2013)
4. Dubey, R., Zhou, J., Wang, Y., Thompson, P.M., Ye, J., et al.: Analysis of sampling techniques for imbalanced data: an n = 648 ADNI study. NeuroImage **87**, 220–241 (2014)
5. He, H., Garcia, E.A.: Learning from imbalanced data. IEEE Trans. Knowl. Data Eng. **21**(9), 1263–1284 (2009)
6. Cao, P., Zhao, D., Zaiane, O.: An optimized cost-sensitive SVM for imbalanced data learning. In: Pei, J., Tseng, V.S., Cao, L., Motoda, H., Xu, G. (eds.) PAKDD 2013. LNCS (LNAI), vol. 7819, pp. 280–292. Springer, Heidelberg (2013). doi:10.1007/978-3-642-37456-2_24

7. Weiss, G.: The impact of small disjuncts on classifier learning. Ann. Inf. Syst. **5**(8), 193–226 (2010)
8. Yuan, M., Lin, Y.: Model selection and estimation in regression with grouped variables. J. Roy. Stat. Soc. B (Stat. Methodol.) **68**(1), 49–67 (2006)
9. Beck, A.: A fast iterative shrinkage-thresholding algorithm for linear inverse problems. SIAM J. Imaging Sci. **2**, 183–202 (2009)
10. Liu, F., Zhou, L., Shen, C., Yin, J.: Multiple Kernel learning in the primal for multimodal Alzheimers disease classification. IEEE J. Biomed. Health Inform. **18**(3), 984–990 (2014)
11. Hinrichs, C., Singh, V., Peng, J., Johnson, S.: Q-mkl: matrix-induced regularization in multi-kernel learning with applications to neuroimaging. In: Advances in Neural Information Processing Systems, pp. 1421–1429 (2012)
12. Gu, B., Sheng, V.S.: A robust regularization path algorithm for ν-support vector classification. IEEE Trans. Neural Netw. Learn. Syst. (99), 1–8 (2016)
13. Gu, B., Sheng, V.S., Wang, Z., Ho, D., Osman, S., Li, S.: Incremental learning for ν-support vector regression. Neural Networks **67**, 140–150 (2015)
14. Ye, J., Liu, J.: Sparse methods for biomedical data. ACM Sigkdd Explor. Newsl. **14**(1), 4–15 (2012)
15. Maldonado, S., Weber, R., Famili, F.: Feature selection for high-dimensional class-imbalanced data sets using Support Vector Machines. Inf. Sci. **286**, 228–246 (2014)
16. Yuan, L., Liu, J., Ye, J.: Efficient methods for overlapping group lasso. In: Advances in Neural Information Processing Systems, pp. 352–360 (2011)
17. Liu, J., Ye, J.: Moreau-Yosida regularization for grouped tree structure learning. In: Advances in Neural Information Processing Systems, pp. 1459–1467 (2010)
18. Figueiredo, M.A.T., Jain, A.K., Doi, K.: Unsupervised learning of finite mixture models. IEEE Trans. Pattern Anal. Mach. Intell. **24**(3), 381–396 (2002)
19. Nesterov, Y.: Smooth minimization of non-smooth functions. Math. Program. **103**(1), 127–152 (2005)
20. Wan, J., Zhang, Z., Rao, B.D., Fang, S., Yan, J., Saykin, A.J., Shen, L.: Identifying the neuroanatomical basis of cognitive impairment in Alzheimer's disease by correlation-and nonlinearity-aware sparse Bayesian learning. IEEE Trans. Med. Imaging **33**(7), 1475–1487 (2014)
21. Weiner, M.W., Aisen, P.S., Jack, C.R., Jagust, W.J., et al.: The Alzheimer's disease neuroimaging initiative: progress report and future plans. Alzheimers Dement. **6**, 202–211 (2010)
22. Ye, J., Farnum, M., Yang, E., Verbeeck, R., et al.: Sparse learning and stability selection for predicting MCI to AD conversion using baseline ADNI data. BMC Neurol. **12**(46), 1–12 (2012)
23. Lebedev, A.V., Westman, E., Van Westen, G.J.P., Kramberger, M.G., et al.: Random Forest ensembles for detection and prediction of Alzheimer's disease with a good between-cohort robustness. NeuroImage: Clin. **6**, 115–125 (2014)

Benchmarking Post-processing Techniques for Offline Arabic Text Recognition System

Sana Khamekhem Jemni[1,2(✉)], Yousri Kesentini[1,2], and Slim Kanoun[1,2]

[1] MIRACL Laboratory, University of Sfax, Sfax, Tunisia
sana.khamekhem@gmail.com, yousri.kessentini@gmail.com, slim.kanoun@gmail.com
[2] Digital Research Center of Sfax, Sfax, Tunisia

Abstract. Automatic recognition of offline Arabic text still faces a big challenge due to the Arabic script nature. Recently, researcher's attention has been increased and variant methods had been applied in this area. This paper presents a comparative study of four OCR (Optical Character Recognition) post-processing error correction techniques. We evaluate their impact using two recognition approaches: a lexicon driven approach with and without the presence of OOV (Out Of Vocabulary) words and a lexicon free-based approach. An AOCR (Arabic Optical Character Recognition) is developed for this purpose. This system is based on HMM (Hidden Markov Model) segmentation free approach. A sliding window is performed on the line image from right to left in order to extract the oriented gradient histogram (HOG) features. Experiments are carried out on KAFD database using different scenarios and revealed a significant improvement in OCR error correction rate.

Keywords: AOCR · HMM · Lexicon · Language model · Post-processing · Sequence alignment

1 Introduction

The Arabic text recognition remains a challenging task due to the desire of the man-machine communication improvement. Research works on this area are becoming more intensive the two last decades and many AOCR systems are available. Information retrieval (IR) becomes an open area for research. The need to convert image documents to editable documents that can be processed, modified, searched and indexed for an unrestricted number of times without any loss of information becomes a challenging task. Often, Errors are committed by OCR systems. Actually, the performance of an AOCR system is far from ideal and recognition errors considerably degrade the performance of NLP (Natural Language Processing) applications and IR systems. During the OCR process, characters in the input document may be substituted with other characters, often ones that are similar in appearance. These errors have an opposite effect on the effectiveness of IR algorithms.

Most widely used OCR post-processing methods [1] are those based on probability and Statistical models, dictionary and mixed methods. The dictionary based

A. Abraham et al. (eds.), *Proceedings of the 16th International Conference on Hybrid Intelligent Systems (HIS 2016)*, Advances in Intelligent Systems and Computing 552,
DOI 10.1007/978-3-319-52941-7_27

post-processing method consists on finding the correct form of a word. A search through all dictionary entries is performed to find the best candidate corrections. This process is frequently used in various applications such as spell-checkers and handwriting recognition. While this technique looks at correcting isolated words, errors related to the context of the sentence are ignored. In effect, context-based correction techniques were proposed to correct OCR errors.

Language modeling (LM) techniques are widely used as post-processing process to correct OCR output errors. It consists on detecting errors by examining all sequences of n words (word n-grams) within a sentence. Dictionaries are constructed with limited vocabularies. However, the LM should be able to create infinite dictionary corrections efficiently. Therefore, an accurate method with a capability of predicting unknown tokens is required. Statistical LM technique is based on n-grams [2, 3]. Other methods are based on finite state machines [4] where the output string is generated from candidate strings having the highest probability of transition. Another way of correcting context-based errors is by constructing rules. The effectiveness of this method was proven on [5].

Voting techniques [12] are an alternative when multi OCR outputs are available. The main idea of this method is to construct an OCR output with the lowest error rate. It consists on minimizing the transition network cost of words via application of dynamic programming.

In this paper, we present a comparative study of a variant of post-processing techniques based on statistical LM, sequence alignment and dictionaries. We study the impact of the different methods depending on the adapted approach. Hence, we evaluate the suitable strategy for the post-processing stage, which is adequate to the both used approach namely lexicon driven (LD) and lexicon free approach (LF).

This paper is organized as follows: The next section presents the Arabic text recognition system used in the evaluation of post-processing strategies. We present a classification of OCR errors in Sect. 3. Experiments and analysis are shown in Sect. 4. In Sect. 5, we present the conclusions of this study.

2 Optical Character Recognition System

AOCR systems have multiple intermediate steps in order to convert a text image into its adequate transcription or Unicode. A segmentation stage is often required to separate characters of a text line image, these fragments were analyzed and sent to the correspondent classifier to be recognized. However, segmentation free approach bypasses this step. A sliding window is scanning the line image from right to left to extract features, which are the input of the HMM classifier. This approach has been proven a success in recognizing a variety of documents without any segmentation requirement. The block diagram of the AOCR system is presented in Fig. 1.

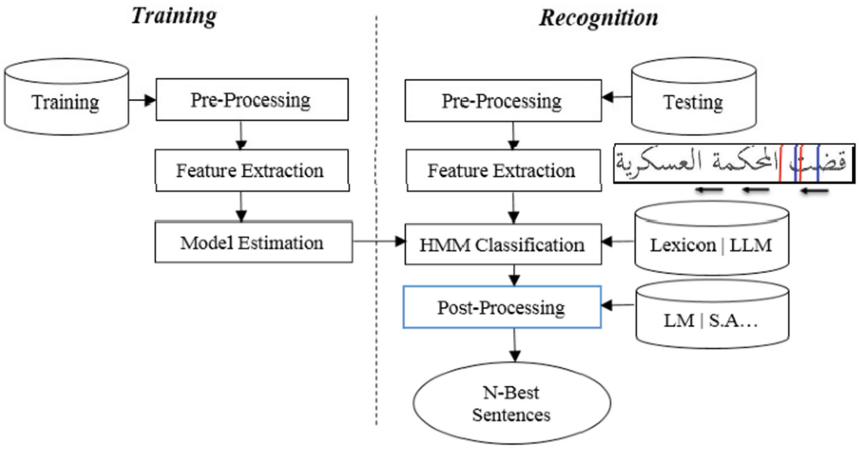


Fig. 1. The AOCR block diagram

2.1 Preprocessing

Preprocessing step is necessary for the noise reduction and the elimination of any variability resources introduced during the scanning phase of images [10]. The Sauvola method [7] is applied to binarize the line image. A normalization step is performed in order to reduce the signal inclination to the horizontal. The skew angle is determined via the image contour, then, the image is corrected [6].

2.2 Feature Extraction

HOG features are widely used in computer vision and image processing for the purpose of recognition. They are inspired from the SIFT (Scale Invariant Feature Transform) key point descriptor [8, 9]. These descriptors are implemented by dividing the image into cells, and a histogram of gradient directions or edge orientations is computed for all the pixels within each cell. For feature extraction, a sliding window of a fixed width is used to extract a sequence of frames from the line image. The window has eight pixels width and two overlapping pixels, moving from right to left and at each position, the frame is divided into cells of $4 * 2$. For each such sub-region or cell, the histogram of oriented gradient (HOG) is computed. At the end of the process, all these histograms are combined to generate the final feature descriptor.

2.3 HMM Recognition

HMMs are one of the most greatly used and thriving classifiers for text recognition [1, 2]. They avoid the necessity of segmentation of text line images into small units, such as characters or strokes, which is adapted when using other classifiers. The text recognition system is realized using HMMs. Character models are considered as the

basic models in this approach. The number of hidden states are fixed to two transitions per state. The HMM character structure is a left-to-right structure which is analogous to that's used in speech recognition. In the current system, we are using five emitting states for all characters HMMs with twelve separate Gaussian mixtures (GMM). In the next stage, the Viterbi algorithm is used to initialize individual HMM for the alignment information from the training. Sentence hypothesis were generated by Viterbi decoding using the HTK toolkit [10, 21].

In order to evaluate the influence of post-processing techniques, two approaches were employed. The LD method was performed to aid the decoding process by exploring only the valid words, which belong to the used dictionary. Despite the effectiveness of this method, the system is unable to recognize OOV Words. Nevertheless, when using the LF approach, the lexicon is not used to drive the recognition process. Thus, any sequence of letters can be recognized. However, the problem is that the confusion is very high. A post-processing stage is generally proposed to fix generated errors and to improve the system performance.

In the next section, we are interested on examining post-processing strategies related to the both approaches.

3 Optical Character Recognition System

Post-processing step is the last stage of an OCR system whose object is to correct spelling errors of the generated transcription. Essentially, there are two types of spelling errors: non word errors and real word errors [11]. A non-word error is a word that is recognized by the OCR system and does not belong to the lexicon language. For instance, when "نهاية الدراسة على درجة البكالوريوس" is recognized by the AOCR system as "نهاية الراسة على درجة البكالوريوس„ then "الراسة" is said to be a non-word error because "الراسة" is not defined in the Arabic language. In contrast, a real-word error is a word that is recognized by the OCR system and correspond to an entry of the used lexicon, although, it is grammatically incorrect. As instance, when "مجالات النمو المعرفي والوجداني" is recognized as "مجالات ينمو المعرفي والوجداني", then "ينمو" is a real word error because it syntactically exists in the Arabic lexicon but grammatically incorrect.

Errors correction techniques relied on dictionary lookup [14] has been widely used in the case of non-real word. These methods were generally used on the last level of the recognition process. We present in the next section the different post-processing techniques.

3.1 Sequence Alignment

The sequence alignment technique (SA) is based on the method proposed by David et al. [16, 17]. This method was used to handle noisy OCR by the alignment of multiple OCR outputs from the same scan. This method is evaluated on Latin books. It consists on three stages. The first stage generates pair-wise alignments of the three sentence hypothesis. In fact, one of the three input texts is selected arbitrary to help as a "pivot". This pivot text is separately carried out into alignment with the other two texts.

The second stage builds an alignment of the three texts from the pair-wise alignments. Words are aligned using an edit distance. The reader can refer to the Yalniz and Manmatha research [23]. The last stage incorporates taking the multiple sequence alignment and generating a corrected text. In this work, an inspired alignment technique is proposed. It consists on combining three-line hypothesis from Top1, Top2 and Top 10 produced by our AOCR.

3.2 Rover

System combination is an area in which printed text recognition has drawn stimulation from speech recognition. Most system combination approaches are comparable to ROVER (Recognizer Output Voting Error Reduction), used in speech recognition to let systems to 'vote' on a confusion network of suggestions. The outputs from multiple systems are joined to give a single proposition, in the hope that the combined information from a number of systems will give a perfect result.

To achieve this, the outputs of multiple of OCR systems are combined into a one hypothesis, minimal-cost for word transition network (WTN) is selected via iterative applications of dynamic programming (DP) alignments. The subsequent network is explored by an automatic rescoring or "voting" procedure that picks the output sequence with the lowest score. The combination of the AOCR output can be performed on the 3-best hypothesis sentences list of one system.

3.3 Word n-grams (WB)

A word n-gram is a sequence of n consecutive words in text. The word n-gram technique is a flexible method that can be used to calculate the likelihood that a word sequence would appear [18]. The major used language model for recognition is an n-gram Markov model. Using this method, the candidate correction of a misspelled word might be successfully selected. For example, in the sentence "إن المخبرين عن هذا العمود" the possible corrections for the word "العمود" might be "العمود" and "المعمور". However, using the n-gram method will likely indicate that the word bigram "المعمور هذا" is much more likely than the trigram "هذا العمود" Thus the word "المعمور" is a more likely correction than "العمود". Thus the general equation for the conditional probability of the next word in a sequence would be:

$$P(w_n|w_1, \ldots, w_{n-1}) \approx P(w_n|w_{n-1})$$

Where $w_1, w_2, \ldots, w_{n-1}$ represents word sequence. The Maximum Likelihood Estimation (MLE) is employed to estimate the probabilities. In the denominator, $C(w_{n-1})$ shows the count of bigrams beginning with w_{n-1}. The general formula for estimating probability for a MLE n-gram is:

$$P(w_n|w_{n-1}) = \frac{C(w_{n-1}, w_n)}{C(w_{n-1})}$$

3.4 Lexicon Matching (L)

Dictionary lookup is based on retrieving candidate terms that help to identify corrupted or miss-spelled words. Several methods relied on edit distance and frequency matrix were used in the literature [22]. In this paper, a matching algorithm based on Damerau Levenshtein distance [13, 20] were employed. We generate all possible candidates with an edit distance <=2 (deletes + transposes + replaces + inserts) from the entry term and search them in the dictionary. Correct generated words that exist in the lexicon were ordered following their frequency on the used corpus. We used the top one generated hypothesis in following experiments.

3.5 Google Suggestions (GS)

The post-processing method and algorithm for OCR error correction based on the "did you mean" spelling suggestion feature of Google's online web search engine [14, 15]. The idea centers on using Google's enormous indexed data to detect and correct misspelled words in the OCR output text. The algorithm starts first by the tokenization of the OCR text into several tokens of words. Then, each token is sent as a search query to Google's search engine so that it is treated. In case the query contains a misspelled word, Google will suggest a possible correction via its "did you mean" feature. Consequently, this spelling suggestion is to be considered as a correction for the misspelled query.

4 Experimental Results

In this section, we introduce the baseline experiments on Arabic script. For all our experiments, we measured the Word Recognition Rate (WRR) following speech recognition conventions (i.e., we subtracted the number of deletions, insertions and subtututions, and divided by the total number of words in the provided transcriptions).

The evaluation of the proposed recognition system is performed using the KAFD database image [19]. It consists of 115,068 page level images for all resolutions (28,767 page images per resolution. It contains 40 fonts with 10 different sizes and 4 styles. Only ten popular fonts were used in our experiments. The training set consists of 20 K line images, while 8 K are used in testing. The images are distributed fairly, 2 K images from each font for training and 800 images for testing. The used data covers 4 styles and 10 sizes.

4.1 Performance Evaluation

The recognition system employed the LD approach in preliminary experiments. The vocabulary in the lexicon consists of 10 K words (words presented in training and testing data set). We carried additional experiments using two reduced lexicons where respectively 46% and 75% of words belonging to the test dataset were eliminated from the used lexicon. Next, we will show the impact of different post processing methods on the LF

recognition system; that does recognition at character level instead of word level. Thus, the system is able to recognize any word by recognizing the sequence of its characters.

Table 1 summarizes the obtained results. The first line presents the WRR using the LD approach. A WRR of 81.68% is obtained for the Top 1-best sentence hypothesis. This rate increases by 3.88% when considering the Top 10-best hypothesis. Reducing the vocabulary to 46%, the WRR decreases by 16.33%. This rate is equal to 53.68% when minimizing lexicon entries to 75%.

Table 1. Word recognition rates

Approach	WRR%			
	Top1	Top2	Top3	Top10
Lexicon driven	81.68	83.08	83.98	85.56
Reduced lexicon (46%)	65.55	65.98	66.31	66.98
Reduced lexicon (75%)	53.68	53.62	53.54	53.28
Free lexicon	56.34	57.26	57.69	59.34

As we cannot cover all possible words in a lexicon. Recognition errors due to the OOV words induce neighboring words. Accordingly, we evaluate the impact of the described post-processing techniques in order to overcome this issue.

Under the same experimental conditions, we present results using a bigram of letters during the decoding stage. This bigram (n-gram, n = 2) is performed on the character's sequence. The recognition rate declines significantly by 25.34% when comparing to the LF approach as presented in the third line of this table. This reduction is expected as the system does not using any linguistic knowledge during the decoding stage. It recognizes the image line as a sequence of characters. It is important to note here that the free lexicon approach gives a better result than the LD approach when the number of OOV words increases. In that, a recognition rate equal to 53.68% is achieved when using an OOV rate of 75%, this rate rises with $\sim$3% when adopting the LF approach. In fact, the LD approach is restricted with a limited lexicon even if it is large. Some correct words can appear during the recognition process without belonging the used vocabulary. However, when using the LF approach, the image line is recognized as a sequence of letters and words boundaries were identified using the 'sp' model.

4.2 Post-processing Techniques Evaluation

For comparison purposes, we applied a variant of post-processing techniques (the sequence alignment technique (SA), Word bi-grams (WB), Lexicon Matching (L) and Google Suggestions(GS)) in order to analyze the effect of each one on the different recognition strategies. We analyze the impact of these techniques with and without the presence of OOV words using the LD and LF approaches. Experiment results are outlined in Table 2.

Using the LD approach, the WRR decreases by 2.78% when using the G.S technique as post-processor. The Google data set incorporates a large amount of data

Table 2. Word recognition results with different strategies (NA: not adapted)

Technique	WRR%			
	Lexicon driven	Reduced lexicon (46%)	Reduced lexicon (75%)	Lexicon free
GS	78.7	64.7	51.41	69.12
SA	82.69	68.04	54.97	69.07
WB	82.92	67.40	53.98	NA
SA + WB	84.22	67.84	53.90	NA
SA + L	NA	NA	NA	74.73
SA + L + WB	NA	NA	NA	79.42

collected from the internet. G.S process is founded on the probabilistic n-gram model presented by [24] for predicting the next word in a sequence of words.

Reducing the test vocabulary size, the recognition rate decreases as expected by 1 to 2% when the G.S method were performed. While this rate increases significantly by 12.78% when adopting the LF based approach. The G.S system can propose an alternative correct word for the often made mistakes, misspellings, and keyboarding errors. Moreover, it can handle real word errors depending on the phrase context due to the titanic database of millions of web pages. When using the S.A method, the WRR increases for the three proposed approach, and especially for the LF approach. This technique is interesting in the case of noisy sequence combination.

A statistical language model is performed as shown in the third line of the Table 2. A word bigram model is applied on the generated transcription. Comparing LM method to the tow earlier techniques. The enhancement in the recognition system performance is closest to 1.5% when using a lexicon LD with all word entries in the lexicon. This factor is limited to 0.3% when using reduced lexicon.

A dictionary lookup is needed when performing the recognition process without a lexicon during the decoding phase. So, a considerable rising by 18% is observed while using a lexicon matching method for each generated term. This process is performed on the aligned sequence in order to ameliorate WRR. The lexicon matching method is not needed in the case of the LD approach as the decoding stage is directed by the used vocabulary. This method is able to correct non-word errors but does not handle real word errors. Thus, using a post-processing technique based on context is necessary to overcome this deficiency. A bigram language model is applied on the generated sentence hypothesis (SA + L + BW) for the purpose of enhancing the performance of the LF based system as shown in Fig. 2. Otherwise, a word bigram ought to be applied on a sequence of words included in the lexicon. As it can be seen in the sixth line of the table, WRR increases by a factor of 23% when performing this combination.

Using the aligned transcription generated by the LD approach, we applied a word bigram, the performance increases further by a factor of 3%. When reducing the lexicon to 46% and below the same conditions, this rate increases also by 2% as presented in Fig. 3. Therefore, this combination can resolve partially the issue of OOV words. This method is not adequate when a vocabulary reduction of 75% is performed as shown in the fourth line of the result table. As we have seen, the SA method is more adapted in this case with an improvement of 1.2% on WRR.

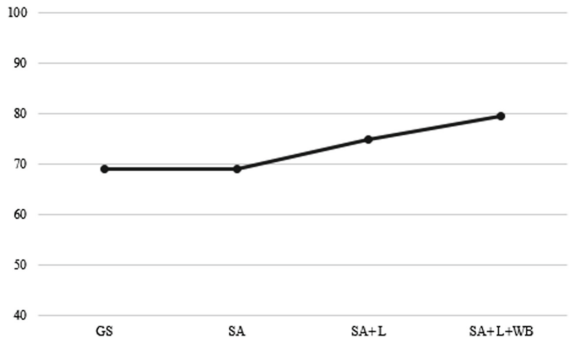


Fig. 2. Post-processing results using the LF approach

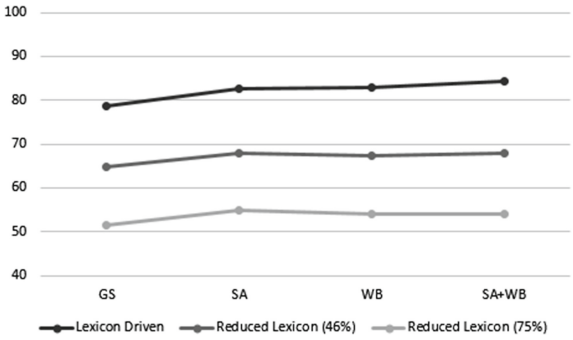


Fig. 3. Post-processing results using the LD approach

These results show a considerable improvement on WRR by doing different post-processing combinations. Three recognition approaches were evaluated accordingly to the used lexicon. The G.S technique is more efficient when using the LF approach. But, WRR degrade when using an approach directed by a lexicon. A method based on sequence alignment is performed. Three sentence hypothesis were combined to generate a new transcription. A relevant improvement on the system performance was observed. We showed that the WB applied on the aligned sequence outperform better than using a LM applied on the top 1-best sentence hypothesis for the LD approach. This method is reliable when some tokens were missing on the used lexicon as the OOV words cases.

5 Conclusion and Future Work

In this paper, a multi-font text recognition system for Arabic script that is based on HMM modeling is presented. Our main contribution consists on the evaluation of different post-processing techniques applied to the described AOCR. Results were developed when applying Google's online suggestion technique on the LF recognition

approach. Using this strategy, the system performance is increased by 12.78% opposed to the use of the LD approach. We have addressed also the issue of OOV words, which can be resolved by using the SA method with an improvement by a factor of 1% to 2%. This strategy allows relevant rising by 23% on the system performance as already shown and especially for the LF approach. LM technique has shown a significant amelioration in WRR for the LD approach. Different strategies compositions have revealed a considerable rising on the WRR. LM interpolation based on morphological decomposition of the Arabic language will be evaluated in later works.

References

1. Fink, G.A., Zhuang, C., Zhu, L.: A post-processing approach for handwritten Chinese address recognition. J. Chin. Inf. Process. (2006)
2. Farooq, F., Jose, D., Govindaraju, V.: Phrase-based correction model for improving handwriting recognition accuracies. Pattern Recogn. **42**(12), 3271–3277 (2009)
3. Perez-Cortes, J., Amengual, J., Arlandis, J., Llobet, R.: Stochastic error correcting parsing for OCR postprocessing. In: International Conference on Pattern Recognition (ICPR), vol. 4, pp. 405–408 (2000)
4. Llobet, R., Navarro-Cerdan, J.R., Perez-Cortes, J.-C., Arlandis, J.: OCR post-processing using weighted finite-state transducers. In: International Conference on Pattern Recognition (ICPR) (2010)
5. Mangu, L., Brill, E.: Automatic rule acquisition for spelling correction. In: International Conference on Machine Learning (ICML) (1997)
6. Hull, J.J.: Documents skew detection: survey and annotated bibliography. In: Document Analysis Systems II, pp. 40–64. World Scientific (1998)
7. Sauvola, J., PietikaKinen, M.: Adaptive document image binarization. Pattern Recogn. (PR) **33**(2), 225–236 (2000)
8. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. Int. J. Comput. Vis. (IJCV) **60**(2), 91–110 (2004)
9. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR), pp. 886–893 (2005)
10. HTK Speech Recognition Toolkit. http://htk.eng.cam.ac.uk/
11. Kukich, K.: Techniques for automatically correcting words in text. ACM Comput. Surv. **24** (4), 377–439 (1992)
12. Fiscus, J.G.: A post-processing system to yield reduced word error rates: recognizer output voting error reduction (ROVER). In: Automatic Speech Recognition and Understanding. National Institute of Standards and Technology, Gaithersburg (1997)
13. Levenshtein, V.I.: Binary codes capable of correcting deletions, insertions, and reversals. Cybern. Control Theor. **10**(8), 707–710 (1966)
14. http://www.googleguide.com/spelling_corrections.html
15. Brants, T., Franz, A.: Web 1T 5-gram Version 1. Linguistic Data Consortium, Philadelphia (2006)
16. Wemhoener, D., Yalniz, I.Z., Manmatha, R.: Creating an improved version using noisy OCR from multiple editions. In: International Conference on Document analysis and Recognition (ICDAR) (2013)
17. Zeki Yalniz, I., Manmatha, R.: A fast alignment scheme for automatic OCR evaluation of books. In: International Conference on Document analysis and Recognition (ICDAR) (2011)

18. Brakensiek, A., Willett, D., Rigoll, G.: Unlimited vocabulary script recognition using character n-grams. In: Proceedings of the 22nd DAGM Symposium, pp. 436–443 (2000)
19. Luqman, H., Mahmoud, S.A., Awaida, S.: KAFD Arabic font database. Pattern Recogn. **47** (6), 2231–2240 (2014)
20. Damerau, F.J.: A technique for computer detection and correction of spelling errors. Commun. ACM **7**, 171–176 (1964)
21. Young, S.J., Evermann, G., Hain, T., Kershaw, D., Moore, G., Odell, J., Ollason, D., Povey, D.,Valtchev, V., Woodland, P.: The HTK Book (for HTK Version 3.2. 1). Cambridge University Engineering Department (2002)
22. Liu, L.-M., Babad, Y.M., Sun, W., Chan, K.-K.: Adaptive post processing of OCR text via knowledge acquisition. In: Proceedings of the 19th Annual Conference on Computer Science (1991)
23. Yalniz, I.Z., Manmatha, R.: A fast alignment scheme for automatic OCR evaluation of books. In: International Conference on Document analysis and Recognition (ICDAR) (2011)
24. Markov, A.A.: Essai d une Recherche Statistique Sur le Texte du Roman. "Eugène Oneguine", Bulletin de l'Académie Impériale des Sciences de St.-Pétersbourg. VI série, 7 (3), 153–162 (1913)

An Empirical Study of the Multi-fragment Tour Construction Algorithm for the Travelling Salesman Problem

Mehdi El Krari[1,2(✉)], Belaïd Ahiod[1,3], and Bouazza El Benani[1,2]

[1] Faculty of Science, Mohammed V University in Rabat, Rabat, Morocco
mehdi@elkrari.com, ahiod@fsr.ac.ma, elbenani@hotmail.com
[2] Computer Science Laboratory, Mohammed V University in Rabat, Rabat, Morocco
[3] LRIT, Associated Unit to CNRST (URAC 29),
Mohammed V University in Rabat, Rabat, Morocco

Abstract. This paper proposes a detailed study of the Multi-Fragment (MF) tour construction (TC) algorithm for the Travelling Salesman Problem (TSP). This TC heuristic is based on an edge selection strategy which favors edges with the smallest cost under the constraint to have a feasible tour. Extensive computational experiments have been performed on benchmark instances from the literature. The results show that the studied algorithm generally outperforms other constructive heuristics for the TSP.

Keywords: Travelling Salesman Problem · Edge selection strategy · Tour construction · Heuristic

1 Introduction

The Travelling Salesman Problem (TSP) is one of the most studied problems in combinatorial optimization [15,19]. It was first introduced by William Rowan Hamilton (1859). Formally, the problem can be defined as follows. Given a finite number of cities and the cost of travel between each pair of them, find the cheapest way of visiting all of the cities and returning to the starting point. It is well-known that the TSP is NP-hard [7]. Since only relatively small instances of the TSP can be solved to optimality within a reasonable amount of time. The problem has been mainly tackled using heuristic approaches, including constructive heuristics, local search heuristics, and metaheuristics [6,11,16].

There are several practical contexts where the TSP is a subproblem as in logistics, transportation of goods, shipping, and many other scheduling problems. The manufacturing of VLSI chips and X-ray crystallography are just a few examples of several issues in the industry that are modelled as a travelling salesman problem. In addition to its practical importance, the TSP is also a standard testbed for new algorithmic ideas.

A. Abraham et al. (eds.), *Proceedings of the 16th International Conference on Hybrid Intelligent Systems (HIS 2016)*, Advances in Intelligent Systems and Computing 552,
DOI 10.1007/978-3-319-52941-7_28

As mentioned earlier, several heuristics have been developed to solve the TSP. These include, among others, k-opt [6], genetic algorithms [8], tabu search [13], simulated annealing [10] and Lin-Kernighan [11]. These heuristics need a starting tour to work on and improve it. They are called tour improvement (TI) heuristics. This tour can be constructed either randomly, by choosing arbitrarily a city at each iteration, or by carrying out so-called tour construction (TC) heuristics, which builds tours according to specific criteria. This paper gives a detailed study of the greedy tour construction algorithm [11], also called the Multi-Fragment (MF) heuristic [2,3], which is based on an edge selection strategy. The purpose is to build a tour with the smallest edges which are already sorted, under the constraint that the tour must be feasible, which makes the MF heuristic different from most known TC methods.

The remainder of this paper is organized as follows: Sect. 2 gives a brief overview of some tour construction heuristics proposed in the literature. The MF heuristic is described in Sect. 3. The computational results on a large set of standard instances from TSPLIB are presented and discussed in Sect. 4. Finally, some conclusions are drawn in Sect. 5.

2 Tour Construction: State of the Art

A TC heuristic [12] has a well-defined rule(s) to construct a feasible tour but makes no improvement once the tour is constructed. In other words, the tour is iteratively built without making changes on parts already built. In what follows, we cite some TC methods frequently used in the literature.

2.1 Nearest Neighbor Heuristic (NN)

The nearest neighbor heuristic (NN) [2] is the simplest heuristic dedicated to the TSP. The salesman starts from a city chosen arbitrarily, and goes to the nearest one and so on so forth. Making sure that none of the cities already visited is re-visited. When all the n (which is the instance size) cities are visited, the salesman returns to the starting city. The procedure is summarized in Algorithm 1 below.

Algorithm 1. Nearest Neighbor heuristic

1: Select randomly a city (c)
2: **while** visited cities are less than n **do**
3: Find nearest city (c_n) to (c) from unvisited cities
4: $c \leftarrow c_n$
5: **end while**

NN's complexity is $O(n^2)$. For each starting city, it provides exactly one tour. Therefore, at most n possible tours can be constructed for an instance of size n, as two tours with two different starting cities may be similar.

2.2 Insertion Heuristic (Ins)

The insertion heuristic [2] is based on the insertion of a city in a partial tour (V). There are several rules and criteria for insertion, the simplest one is to select a city among those unvisited (U) and insert it next to the nearest (resp. farthest) city from those existing in V (**Nearest** (resp. **Farthest**) **Insertion**). Once inserted, the selected city is removed from U. Another approach is to insert the city in V so as to minimize the cost of the new sub-tour (**Smallest Sum Insertion**). The process is repeated until U is empty.

Algorithm 2. Insertion heuristic

1: Build a partial tour (V) from two cities
2: **while** U is not null **do**
3: Insert a city (c) from U in the tour V according to the chosen rule
4: $U \leftarrow U \backslash \{c\}$
5: **end while**

There are $(n \times (n-1)/2)$ pairs of cities (edges) to start the construction of the tour. The number of tours provided by the insertion heuristic depends on the selection rule used. For example, if the nearest insertion criterion is used, the insertion heuristic gives 2^{n-2} tours for each couple chosen. While it gives n tours for each starting edge if the Smallest Sum insertion is used. Whatever the rule is, the insertion heuristic's complexity is $O(n^2)$.

2.3 Tour Construction with Spanning Trees

The TSP can be interpreted as a weighted graph $G = (V, E, M)$, where

- V is a set of vertices
- E is a set of edges $\{v, w\}$ where $v, w \in V$
- M is a map from edges to values, which are positive numbers (distances) in the case of TSP.

This TC heuristic is based on the minimum spanning tree (MST) (a subtree $S_G = (V, E')$ (acyclic subgraph) of G, where $E' \subseteq E$) of an instance of TSP [9]. Once the tree is found, the salesman goes around its outside, starting at any node and not yet visited cities along the tree, until he comes back to the starting city. Figure 1 gives an example of visiting cities through a spanning tree: Considering the node (1) as a starting city, the salesman goes around the tree and visits the node (2) then (3). By continuing the tour, the salesman will not visit the node (2) since it has been visited previously and will visit node (4) then (5), and so on.

There are several algorithms to find the MST of a graph (and therefore an instance of TSP):

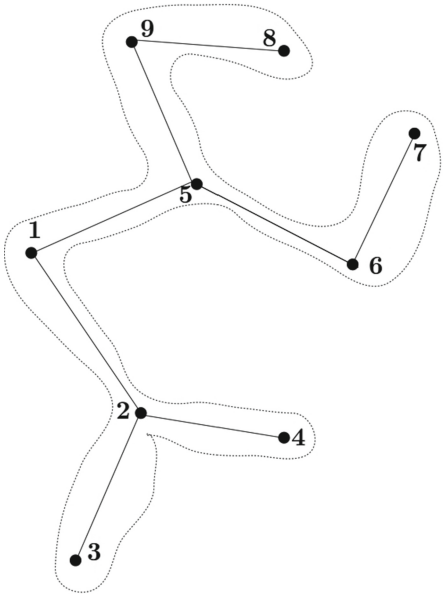


Fig. 1. Going around the Minimum Spanning Tree

- The first algorithm was proposed by Borůvka [4] (1926). It is based on merging disjoint trees. At the beginning of the procedure, each vertex is considered as a separate tree. In each step, the algorithm merges every component with some other using strictly the cheapest outgoing edge of the given component.
- Kruskal's algorithm [14] (1956) creates a forest of trees which consists of n single node trees without edges. At each step, we add the cheapest edge so that it joins two trees together. If it were to form a cycle, it would simply link two nodes that were already part of the same tree, so that this edge would not be needed.
- Prim's algorithm [17] (1956) starts with one node. Then, it sequentially adds a node that is unconnected to the new graph, each time selecting the node whose connecting edge has the smallest weight out of the available nodes connecting edges.

3 Multi-fragment Heuristic

The MF heuristic [2,3] is an interesting approach that considers the edges as the main parameter. The idea is to select edges accordingly to their respective costs. Since the TSP's objective is to find a tour whose cost is minimal, this heuristic aims to select the edges as minimal as possible. Then, the edges are sorted. It is almost sure that the n smaller edges will not give a Hamiltonian feasible tour, and through all the cities of the problem. This heuristic builds a tour by connecting edges while avoiding to:

– Close a tour while unvisited cities exist (see Fig. 2).
– Add an edge of which one of the two cities is already linked to two on the under construction tour. For example, in the scenario in Fig. 3, the algorithm will not choose the edge {b,e} because the city 'b' cannot be linked to more than two cities in a tour.

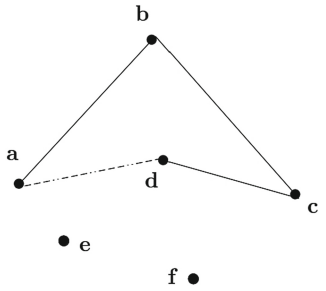


Fig. 2. Closing a tour smaller than n

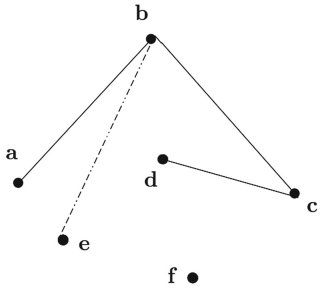


Fig. 3. Linking a city to an already connected city

MF is a deterministic heuristic. For any instance of the TSP, the tour built by this TC heuristic is unique since we choose the first edge found if many exist with the same cost. The results given in the next section show that MF heuristic often goes through a large number of edges to build the appropriate tour. Browsing all edges of the problem makes the complexity of the MF heuristic equal to $O(n^2)$. A pseudo-code of MF heuristic is given in Algorithm 3, while Fig. 4 shows an example of constructing a solution (tour) by MF heuristic, for a TSP instance of 5 cities described by the complete graph in the right. Each row is an iteration of MF heuristic; the first column displays the selected edge. The second column shows the steps of the tour construction: the cities separated by a hyphen are part of the same section, while the pipe separates two sections not yet connected. The last column shows why the edge was rejected.

edges	tour	
1,3	1-3	
2,5	1-3 | 2-5	
2,4	1-3 | 4-2-5	
4,5	1-3 | 4-2-5	(4,5) is closing a tour: rejected
1,2	1-3 | 4-2-5	(2) connected to (4) & (5) : (1,2) rejected
1,4	3-1-4-2-5	

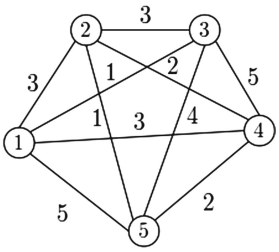


Fig. 4. Example of a tour construction using MF heuristic for an instance of 5 cities

Algorithm 3. MF heuristic

Require: Sorted set of all edges of the problem E.
Ensure: A tour T.

```
for each e in E do
    if (e is closing T and size(T) < n) or (e has a city already connected to two others) then
        go to the next edge
    end if
    if e is closing T and size(T) = n then
        add e to T
        return T
    end if
    add e to T
end for
return T
```

4 Experimental Results

We tested MF heuristic on 33 instances of the TSPLIB [18,20], whose size ranges between 51 and 5934. First, we present a detailed MF heuristic performances. Then, we compare these results to those obtained with other TC methods.

MF heuristic is coded in Java 1.7, All tests were run on an Intel(R) Core(TM) i3-4150 CPU @ 3.50 GHz and 4 GB of memory.

4.1 MF Heuristic's Performances

Since MF heuristic gives the same permutation for each run, each instance was solved only once. Table 1 below gives some statistics about MF heuristic results on different instances from the TSPLIB benchmark: each line gives the "Instance" name, "Size", Best Known Solution (BKS) taken from [20], the objective value of the tour constructed — $Cost$, deviation of obtained tour from the BKS with a value from 0% (the best solution is found in this case) — δ (Eq. 1), CPU time needed to build each tour in seconds, and finally the ratio of edges browsed – from all possible edges – before closing the tour with a value up to 100% (meaning that all edges have been checked)— BE (Eq. 2).

$$\delta = 100 \times (C - BKS)/BKS[\%] \tag{1}$$

$$BE = 100 \times BrowsedEdges/(n \times (n-1)/2)[\%] \tag{2}$$

The results in Table 1 indicate the good performance of the MF heuristic, the average of deviation over all of them is only 14.44%. As mentioned in the previous section, MF heuristic browses majority of edges to build the tour, 74% (as an average) of edges are browsed to get a feasible tour. Instances with a low BE are of better quality because the tour is composed of edges with the smallest cost. The Fig. 5 below gives a correlation between standard deviation and browsed edges for the listed instances above and 45 others. It shows that the more edges MF heuristic browses, the larger the value of deviation is.

Table 1. MF heuristic's performances

Instance	Size	BKS	*Cost*	δ (%)	CPU (s)	BE (%)
st70	70	675	750	11.11	0.14	62
kroA100	100	21282	24287	14.12	0.15	54
pr107	107	44303	47545	7.32	0.15	59
pr136	136	96772	116048	19.92	0.15	58
ch150	150	6528	7809	19.62	0.15	90
kroA200	200	29368	34554	17.66	0.15	65
gil262	262	2378	2739	15.18	0.17	68
rd400	400	15281	17272	13.03	0.17	83
fl417	417	11861	12931	9.02	0.16	46
rat575	575	6773	7858	16.02	0.17	62
u1060	1060	224094	258947	15.55	0.23	66
rl1304	1304	252948	285123	12.72	0.25	49
nrw1379	1379	56638	66106	16.72	0.32	77
fl1577	1577	22249	25221	13.36	0.31	97
d1655	1655	62128	72498	16.69	0.36	99
u1817	1817	57201	65361	14.27	0.32	72
d2103	2103	80450	95124	18.24	0.37	100
u2319	2319	234256	260659	11.27	0.34	84
fl3795	3795	28772	32475	12.87	0.63	96
rl5934	5934	556045	634314	14.08	1.38	93

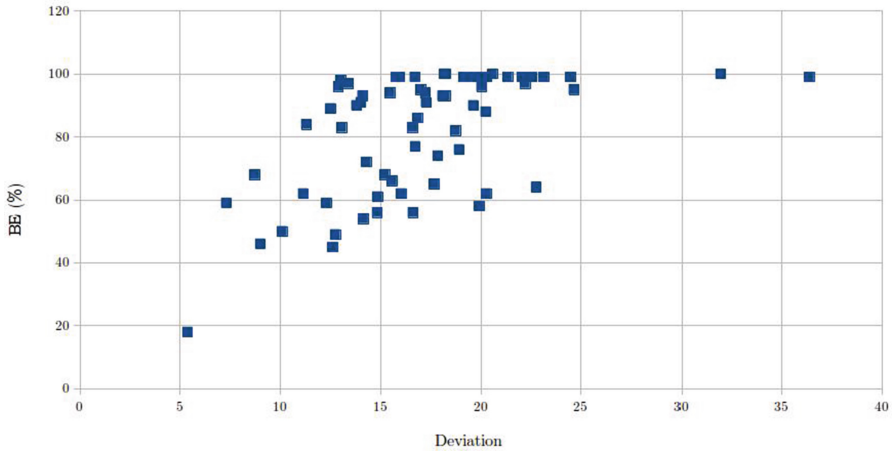


Fig. 5. Correlation between standard deviation and browsed edges of each instance

4.2 Comparing MF Heuristic with Other TC Methods

Table 2 below compares MF heuristic and the following TC methods, which has been described in Sect. 2: (a) the Farthest Insertion (FI); (b) the Nearest Neighbor (NN); (c) Borůvka's algorithm (Bor); (d) the Quick-Borůvka algorithm (Q-Bor). Since these methods can provide different solutions, each of them has been run 20 times with different starting cities/edges. We report the average deviation (δ), as defined previously, average over the 20 runs, and the average CPU time in seconds ($CPU(s)$). Recall that, as mentioned previously, when using the MF heuristic, the instances were solved only once. Note also that while MF heuristic is coded in Java, the other methods used in the comparison are from the Concorde framework [1] and thus written in C, lighter than Java [5]. Running CPU time is given only for information.

Table 2. Comparing MF heuristic to other TC methods

Instance	Size	BKS	FI ($\delta, CPU(s)$)		NN ($\delta, CPU(s)$)		Q-Bor ($\delta, CPU(s)$)		Bor ($\delta, CPU(s)$)		MF ($\delta, CPU(s)$)	
st70	70	675	27.96	0.06	22.83	0.05	23.35	0.06	12.15	0.06	**11.11**	0.05
kroA100	100	21282	26.85	0.06	26.17	0.05	28.47	0.06	19.57	0.06	**14.12**	0.06
pr107	107	44303	**5.18**	0.06	14.63	0.06	20.46	0.06	7.15	0.06	7.32	0.05
pr136	136	96772	22.18	0.07	24.14	0.06	21.99	0.07	**19.92**	0.07	**19.92**	0.06
ch150	150	6528	24.86	0.07	20.13	0.06	25.87	0.06	22.43	0.07	**19.62**	0.06
kroA200	200	29368	35.48	0.06	27.72	0.06	20.89	0.06	18.53	0.06	**17.66**	0.06
gil262	262	2378	31.96	0.06	23.57	0.06	20.15	0.06	15.33	0.06	**15.18**	0.06
rd400	400	15281	34.90	0.07	25.98	0.06	18.05	0.06	22.48	0.06	**13.03**	0.07
fl417	417	11861	31.72	0.06	30.79	0.06	37.13	0.06	23.85	0.06	**9.02**	0.07
rat575	575	6773	36.57	0.06	26.22	0.06	18.22	0.06	17.26	0.06	**16.02**	0.07
u1060	1060	224094	38.86	0.06	32.34	0.06	17.82	0.06	17.41	0.06	**15.55**	0.11
rl1304	1304	252948	46.85	0.06	27.68	0.06	19.00	0.06	14.69	0.06	**12.72**	0.11
nrw1379	1379	56638	39.09	0.06	24.44	0.06	16.67	0.06	**15.22**	0.06	16.72	0.14
fl1577	1577	22249	38.17	0.06	24.15	0.06	30.16	0.06	17.95	0.06	**13.36**	0.16
d1655	1655	62128	45.10	0.06	22.36	0.06	19.15	0.06	**15.19**	0.06	16.69	0.14
u1817	1817	57201	47.00	0.06	24.01	0.07	16.95	0.06	15.43	0.06	**14.27**	0.17
d2103	2103	80450	52.56	0.06	13.68	0.07	**8.52**	0.06	12.97	0.06	18.24	0.18
u2319	2319	234256	35.85	0.06	20.49	0.07	12.98	0.06	15.05	0.06	**11.27**	0.18
fl3795	3795	28772	45.02	0.07	29.34	0.10	27.62	0.07	19.02	0.07	**12.87**	0.44
rl5934	5934	556045	51.61	0.07	22.33	0.13	15.11	0.07	**13.68**	0.07	14.08	0.66
Average			35.89	0.06	24.15	0.06	20.93	0.06	16.76	0.06	**14.44**	0.14

Results in Table 2 show that the MF heuristic is very competitive. For the 20 instances considered in this paper, MF heuristic gives the best tour in 15 instances, as opposed to Borůvka and Quick-Borůvka, which provide the best solutions only in 4 and 1 instances, respectively.

The TC heuristics based on edges are more effective than others which select nodes since the choice of each edge/distance affects directly the quality of the tour.

5 Conclusion

This paper proposed a detailed and empirical study of the MF heuristic designed to build a tour for the travelling salesman problem. This heuristic is based on the selection of edges in an ascending order to build the tour with smaller edges. The results of the MF heuristic on multiple instances of different size of TSP instances demonstrated its effectiveness.

Future research will be devoted to adapt this heuristic to solve other variants of the TSP such as the generalized and asymmetric TSP.

References

1. Applegate, D., Bixby, R., Chvátal, V., Cook, W.: Concorde: a code for solving traveling salesman problems (1999). http://www.math.uwaterloo.ca/tsp/concorde.html
2. Bentley, J.J.: Fast algorithms for geometric traveling salesman problems. ORSA J. Comput. **4**(4), 387–411 (1992)
3. Bentley, J.L.: Experiments on traveling salesman heuristics. In: Proceedings of the First Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 1990, pp. 91–99. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA (1990). http://dl.acm.org/citation.cfm?id=320176.320186
4. Borůvka, O.: O jistém problému minimálním (about a certain minimal problem) (in Czech, German summary) (1926)
5. Bull, J.M., Smith, L.A., Pottage, L., Freeman, R.: Benchmarking java against c and fortran for scientific applications. In: Proceedings of the 2001 Joint ACM-ISCOPE Conference on Java Grande, pp. 97–105. ACM (2001)
6. Croes, G.A.: A method for solving traveling-salesman problems. Oper. Res. **6**(6), 791–812 (1958). http://dx.doi.org/10.1287/opre.6.6.791
7. Garey, M.R., Johnson, D.S., Stockmeyer, L.: Some simplified np-complete problems. In: Proceedings of the Sixth Annual ACM Symposium on Theory of Computing, STOC 1974, pp. 47–63, NY, USA (1974). http://doi.acm.org/10.1145/800119.803884
8. Grefenstette, J., Gopal, R., Rosmaita, B., Van Gucht, D.: Genetic algorithms for the traveling salesman problem. In: Proceedings of the First International Conference on Genetic Algorithms and their Applications, pp. 160–168. Lawrence Erlbaum, New Jersey (1985)
9. Held, M., Karp, R.M.: The traveling-salesman problem and minimum spanning trees. Oper. Res. **18**(6), 1138–1162 (1970)
10. Helsgaun, K.: An effective implementation of the lin-kernighan traveling salesman heuristic. Eur. J. Oper. Res. **126**(1), 106–130 (2000)
11. Johnson, D.S., McGeoch, L.A.: The Traveling Salesman Problem: a case study in local optimization. Local Search Comb. Optim. **1**, 215–310 (1997)
12. Johnson, D.S., McGeoch, L.A.: Experimental analysis of heuristics for the STSP. In: Gutin, G., Punnen, A.P. (eds.) The Traveling Salesman Problem and its Variations, pp. 369–443. Springer, Heidelberg (2007)
13. Knox, J.: Tabu search performance on the symmetric traveling salesman problem. Comput. Oper. Res. **21**(8), 867–876 (1994)
14. Kruskal, J.B.: On the shortest spanning subtree of a graph and the traveling salesman problem. Proc. Am. Math. Soc. **7**(1), 48–50 (1956)

15. Lawler, E.L., Lenstra, J.K., Kan, A.R., Shmoys, D.B.: The Traveling Salesman Problem: A Guided Tour of Combinatorial Optimization, vol. 3. Wiley, New York (1985)
16. Mladenović, N., Hansen, P.: Variable neighborhood search. Comput. Oper. Res. **24**(11), 1097–1100 (1997)
17. Prim, R.C.: Shortest connection networks and some generalizations. Bell Syst. Techn. J. **36**(6), 1389–1401 (1957)
18. Reinelt, G.: TSPLIB–a traveling salesman problem library. ORSA J. Comput. **3**(4), 376–384 (1991). http://dx.doi.org/10.1287/ijoc.3.4.376
19. Reinelt, G.: The Traveling Salesman: Computational Solutions for TSP Applications. Springer, Heidelberg (1994)
20. Reinhelt, G.: {TSPLIB}: a library of sample instances for the TSP (and related problems) from various sources and of various types (2014). http://comopt.ifi.uniheidelberg.de/software/TSPLIB95

Building Probabilistic Ontologies Based on Meta-Model PODM

Hlel Emna(✉), Jamoussi Salma, Turki Mohamed, and Ben Hamadou Abdelmajid

Miracl Laboratory, Technology Center of Sfax BP 242, 3021 Sakiet Ezzit, Sfax, Tunisia
emnahlel@gmail.com, med.turki@gmail.com, {salma.jamoussi,abdelmajid.behamadou}@isimsf.rnu.tn

Abstract. In (Hlel et al. 2016), we have presented an extension of OWL2 meta-model, called Probabilistic Ontology Definition Meta-model (PODM), for representing the fundamental elements of probabilistic ontologies (POs). Indeed, we have presented a list of new probabilistic components which allow representing the probabilistic basic elements of a domain of interest. PODM can be used by users for creating probabilistic ontologies of complex domains. In this article, we will present how we can construct probabilistic ontologies based on this meta-model.

Keywords: Probabilistic ontology · Meta-model · OWL2 · PODM · Probabilistic assertion · Uncertainty

1 Introduction

In literature, various researchers address the requirements to model the uncertain and probabilistic knowledge in the semantic web. The authors of (Predoiu and Stuckenschmidt 2010) have described some areas where probabilistic information plays a role in the context of the semantic web such as representation of uncertain information, ontology learning, etc. None of the existing ontologies languages such as RDF/RDFS, SHOE, OWL provide a means for representing this knowledge. Different probabilistic approaches for extending these languages, especially OWL, with the ability to support uncertainty are explored in literature (Yang 2007; Salvatore 2015). However, currently there is no established foundation or no standard for doing so. Moreover, these works have not focused on the proposal of a meta-model for defining the fundamental components of probabilistic ontologies which allow representing the probabilistic knowledge.

On the other hand, there are various meta-models in the literature for defining the components of classical ontologies (which allow representing the deterministic (classical) knowledge) like W3C OWL2 meta-model (ODM: Ontology Definition Meta-model for OWL2) (Motik et al. 2012). However, the components of POs (which allow representing the probabilistic and uncertain knowledge) are not taken into account. In (Hlel et al. 2016), we have presented an extension of OWL2 meta-model, called PODM, for representing the fundamental elements of probabilistic ontologies. Indeed, we have presented a list of new probabilistic components which allow

A. Abraham et al. (eds.), *Proceedings of the 16th International Conference on Hybrid Intelligent Systems (HIS 2016)*, Advances in Intelligent Systems and Computing 552,
DOI 10.1007/978-3-319-52941-7_29

representing the probabilistic basic elements of a domain of interest. PODM can be used by users for creating POs. In this article, we will present how we can construct a probabilistic ontology of a particular domain based on this meta-model (PODM).

This article is organized as follows. In Sect. 2, we begin with a description of some related works. Next, we introduce our Probabilistic Ontology Definition Meta-model (PODM). In Sect. 4, we present a list of new probabilistic elements for supporting the uncertainty to Assertion. Then, we present our method for constructing probabilistic ontologies based on PODM. Finally, we finish with a conclusion and perspectives.

2 Related Work

The uncertainty is a ubiquitous aspect of most real world problems. It exists in almost every aspects of ontology engineering (Ding and Peng 2004). Today, there is a very interesting requirement to develop formalisms of knowledge representation allowing to deal with uncertainty. Various researchers address the need to model the probabilistic and uncertain information in the semantic web. The authors of (Predoiu and Stuckenschmidt 2010) describe five areas where probabilistic information plays a role in the context of the Semantic Web: Ontology Learning, Ontology Mapping Usage for Information Integration, Representing inherently uncertain Information, Ontology Matching and Document Classification. Despite most researchers have focused on the representation of uncertainty in ontologies; however, currently there is not established foundation or standard for doing so. In literature, there are different approaches for modeling the uncertainty in ontologies. These approaches present various extensions of Description logics and extensions of the web semantic languages for representing uncertain and probabilistic knowledge (Ding 2005; Yang and Calmet 2005; Fabio et al. 2011). None of the existing semantic web languages such as RDF/RDFS, SHOE and OWL provide a means for representing uncertain and probabilistic knowledge of real world domains. Different probabilistic approaches for extending these languages, especially OWL, with the ability to support uncertainty are explored in literature. Indeed, several Bayesian-based approaches to model uncertainty in ontologies have been proposed: BayesOWL (Ding and Peng 2004), OntoBayes (Yang 2007) and PR-OWL (Costa and Laskey 2006). BayesOWL (Ding and Peng 2004) is one proposal to represent the uncertainty in OWL ontologies through Bayesian network (BN) (Ben Mrad et al. 2015; Finn 1996). Probabilistic OWL (PR-OWL) (Costa and Laskey 2006) is a probabilistic ontology approach that is implemented on the basis of first-order logic. It is a probabilistic extension which enables OWL ontologies to represent MEBNs (Multi-Entity Bayesian Networks) (Laskey 2008). It provides a number of new OWL constructs for constructing POs probabilistic ontologies. OntoBayes (Yang 2007) is an ontology-driven uncertainty model, which integrates Bayesian network into OWL ontologies for preserving their advantages. It was developed as an extension which enables OWL ontologies to represent BNs. Indeed, the authors of (Yang 2007) have proposed an upper ontology, called Ontology OntoBayes, for representing random variables of Bayesian network, dependencies between them and probabilities associated to these variables. The representation of probabilistic knowledge in BayesOWL is performed via additional language markups, which can be simply viewed as an upper ontology (Yang 2007).

Description Logics (DLs) (Baader et al. 2003) are a family of ontological knowledge representation languages. They represent knowledge in terms of objects, concepts, and roles. To encode uncertainty, probabilistic description logics must be contemplated. The literature contains a number of proposals for probabilistic description logics (Giugno and Lukasiewicz 2002; Fabio et al. 2011). P-SHOQ is a probabilistic description logic (Giugno and Lukasiewicz 2002), extension of the DL-SHOQ (Horrocks and Sattler 2001). It adds to the syntax for SHOQ a list of conditional constraints that are defined as expressions P(D|C) [l, u] with C, D are classes and [l, u] is an interval between 0 and 1. These constraints can be used to represent different kinds of probabilistic knowledge, for example (D|{o})[l; u] means "*o is an instance of the concept D with a probability in [l; u]*". CRALC (Fabio et al. 2011) is a probabilistic description logic, extension of the DL-ALC (Schmidt-Schauss and Smolka 1991). It retains all constructors offered by ALC (conjunction, disjunction, etc.) by adding probabilistic inclusion such that $P(C \mid D) = \alpha$ *or* $P(r) = \beta$, with C and D are two concepts and r is a role.

In literature, various works have been proposed for representing POs. However, currently there is no established foundation or no standard for doing so. Moreover, these works have not focused on the proposal of a meta-model for defining POs by specifying the new probabilistic components of ontology, which allow representing the uncertainty. We think that future standard OWL2 versions should be extended in a way to allow the creation of the POs. In (Hlel et al. 2016), we have presented an extension of OWL2 meta-model, called PODM, for representing the fundamental elements of probabilistic ontologies. Indeed, we have presented a list of new probabilistic components which allow representing the probabilistic basic elements of a domain of interest. To our knowledge, this work is the first one to propose a meta-model which provides support for defining POs. So, PODM can be used by users for creating POs. In the following, we present this meta-model.

3 PODM: Probabilistic Ontology Definition Meta-Model

In (Hlel et al. 2016), we have presented an extension of OWL2 meta-model, called PODM, for representing the fundamental elements of POs. Indeed, we have presented a list of new components which allow representing the probabilistic basic elements of a domain of interest like *Probabilistic Individual*, *Probabilistic Class*, etc. (see Fig. 1). In the following, we present these components.

Probabilistic Individual. The attribution of data or objects to the corresponding concept (or class) may be uncertain. For example, "*Tom*" is an instance of class "*Animal*" with a probability equal to 0.6 and it is an instance of class "*Person*" with a probability equal to 0.4. This type of instance is called probabilistic or uncertain individual. It is associated with a probabilistic value expressing the belonging degree of an instance to a corresponding concept. Similarity to (Motik et al. 2012), we can distinguish two kinds of probabilistic individual: probabilistic named individual (identified with URI) and probabilistic anonymous individual.

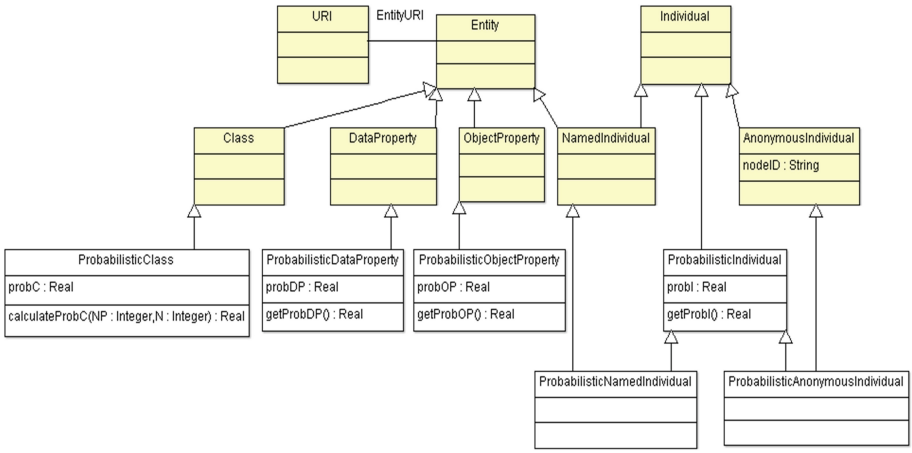


Fig. 1. Probabilistic Ontology Definition Meta-model (PODM). (Note that the classes with color white represent the new probabilistic components and the classes with color yellow represent the classical components of ODM of OWL2.)

Probabilistic class. The classes (or concepts) of OWL ontology describe a collection of objects for a particular domain. If this collection includes one or more probabilistic instances then the type of this class becomes a probabilistic class. Let C be a class of an OWL ontology and $I = \{I_1, \ldots, I_i, \ldots, I_n\}$ be a list of instances of this concept. In OWL ontology, we can distinguish two types of concepts: if all elements of I are classical instances then C is a classical concept and if I contains at least one probabilistic instance then C is a probabilistic concept. Assuming that C is a probabilistic concept, N is the total number of instances of this concept and NP is the number of probabilistic instances of C. This concept is attached with a probabilistic value $ProbV$ which expresses the uncertainty (Hlel et al. 2016):

$$\mathbf{ProbV} = \mathbf{NP}/\mathbf{N} \in]\mathbf{0}, \mathbf{1}] \tag{1}$$

Probabilistic Data Property. Generally, the extraction of knowledge in an automatic way provides us uncertain and undetermined knowledge, because the knowledge extracted by using automatic or semi-automatic systems is uncertain and probabilistic. For example, the extraction of hobby for each person can be realized automatically or semi-automatically from social networks (Facebook, Twitter, etc.). The result of this task is a list of uncertain and probabilistic knowledge (list of hobbies for each person). In an ontology, the concept "*Person*" can be used to model the set of persons. The data property "*name*" can be used to represent the name for each person. The data property "*hobby*" can be used to model the hobbies for each person. The first property is a precise element of this ontology. However, the second property is a probabilistic element of this ontology (probabilistic data property). It is attached with probabilistic value that expresses the degree of certitude of this knowledge. For example, the hobby

of "*John*" (instance of *Person*) is "*music*" (value of the probabilistic data property "*hobby*") with a degree equal to 0.5.

Probabilistic Object Property. In the real world, it is often the case that the relationships between resources hold probabilistically. For example, "*Imagery*" (Theme) is connected to "*Data-Mining*" (Theme) with a probability of 0.7. In PO, this relation ("*be-connected*") is considered as probabilistic object property which is associated with a probability. R is probabilistic object property between two instances if and only if it represents a probabilistic interaction between these two components.

4 Extending PODM with a Probabilistic Assertion

OWL2 supports a rich set of axioms for stating assertions (axioms about individuals that are often also called facts). In this paper, we are concentrated only to these assertions: *ClassAssertion*, *ObjectPropertyAssertion* and *DataPropertyAssertion*. We have extended PODM with a list of new probabilistic element named probabilistic Assertion (see Fig. 2) and its sub-classes (*ProbabilisticClassAssertion*, *ProbabilisticObjectPropertyAssertion* and *ProbabilisticDataPropertyAssertion*) for attaching the uncertainty to Assertion. The *ClassAssertion* axiom allows one to state that an individual is an instance of a particular class. The new axiom *ProbabilisticClassAssertion*

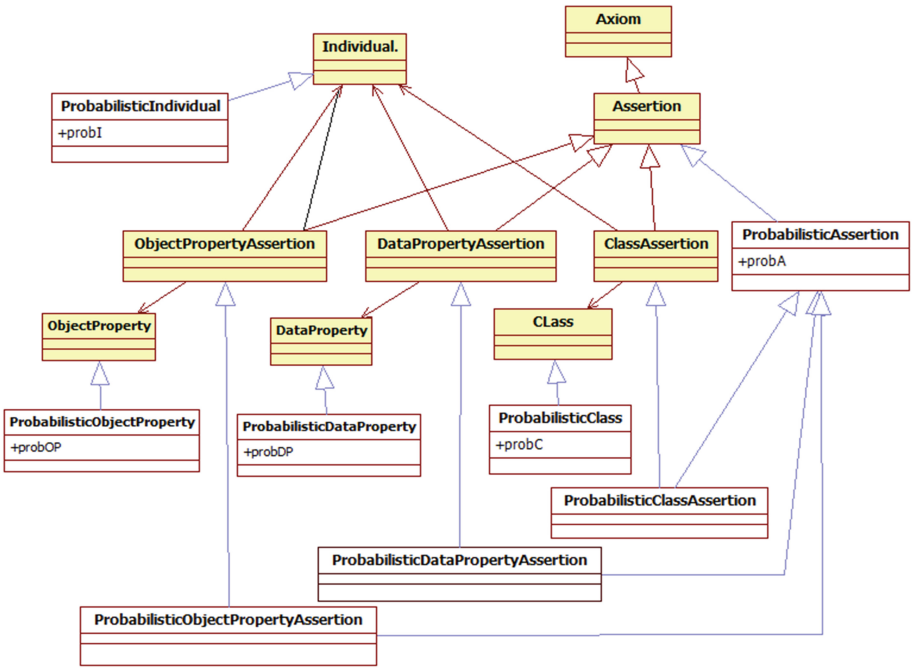


Fig. 2. Probabilistic Assertion. Note that the classes with color white represent the new probabilistic components and the classes with color yellow represent the classical components of ODM of OWL2.

allows to state that a probabilistic individual is an instance of a class with a particular probability. The *ObjectPropertyAssertion* axiom allows one to state that an individual is connected by an object property expression to an individual. The *ProbabilisticObjectPropertyAssertion* allows to state that an individual is connected by a probabilistic object property expression to an individual with a probabilistic value for example "john" is interested to "Films" with a probability of 0.9. The *DataPropertyAssertion* axiom allows one to state that an individual is connected by a data property expression to literal. The *ProbabilisticDataPropertyAssertion* axiom allows to state that an individual is connected by a probabilistic data property expression to literal for example; "Smith" prefers the hobby "travel" with a probability of 0.5 and prefers "music" with a probability of 0.4.

5 Constructing a Probabilistic Ontology by Using PODM

During the past years, the ontologies are widely used for representing knowledge of most real world domains. They provide a definition of concepts, relationships, and other features related to modeling knowledge of particular domain (Gruber 1995). Thanks to these elements, they are used to model the reality (real world applications). However, this world includes inaccuracies and imperfections which cannot be represented by classical or traditional ontologies (COs). For allowing agents to deal with uncertainty, an extension of ontologies which has the capability of supporting uncertain and probabilistic knowledge is mandatory. POs have come to remedy this defect (Costa and Laskey 2006). We can define the PO simply as a CO enriched with uncertain and probabilistic knowledge. Indeed, POs augment COs with the ability to represent the uncertainty (Hlel et al. 2015; Hlel et al. 2014). A reader interested by CO can find various works describing in detail the process of construction of ontology (CO), its components, various automatic, semi-automatic or manual construction methods of CO, etc. (Gómez-Pérez et al. 2006; Stephen and Adam 2015). However, this is not available for PO such as the majority of researchers have focused only on the proposal of extensions of probabilistic description logics and languages of semantic web to model the uncertain knowledge of a particular field (Yang 2007; Costa and Laskey 2006).

In this section, we propose a new method to guide the users for constructing POs based on PODM which provides support for defining POs. This method includes these phases: Specification of requirements, Identification and description of certain (deterministic) knowledge and uncertain (probabilistic) knowledge of domain of interest and Construction of probabilistic ontology by using a formal language. In the following, we will present these phases.

Specification of requirements. This step determines the domain and the purpose of ontology: It is important to be clear identified the purpose (goal) of the ontology. In addition, the ontologist must verify the necessity of the creation of PO through research of uncertainties and inaccuracies in the field of study. For more explaining our method of construction of a PO by using PODM, we have tried to build a probabilistic ontology, named O, which describes a list of peoples as well as their preferences (animal, music, etc.). We assume that these preferences are determined by an automatic

system from social network based on techniques of text mining and natural language processing. Generally, these techniques us provide probabilistic and uncertain knowledge. In our case, the discovered preferences are considered as probabilistic knowledge. So, it is necessary to construct a probabilistic ontology for describing a list of people as well as their preferences.

Identification of probabilistic and deterministic knowledge. In this step, we have determined and described the knowledge of domain of interest by specifying the main probabilistic components of PO (probabilistic concepts, probabilistic individual, probabilistic proprieties) and their characteristics. Moreover, it is necessary identify and

Table 1. Description of components (probabilistic and deterministic) of the ontology O.

Components of ontology		Description of components
Classical Classes	Interest	This classical concept expresses the interests of different people
Classical DataProperties	Name	The class Person is characterized with two Data Properties: name and address
	Address	
	description-Interest	The class Interest is characterized with this property description
Classical ObjectProperties	have-friend	This ObjectProperty expresses a friendly relationship between persons
Probabilistic Classes	Person	This probabilistic concept expresses a list of people. It is probabilistic because it has some probabilistic instances like “tom” and “loulou”
	Animal	This probabilistic concept expresses a list of animals. It is probabilistic because it has some probabilistic instances like “tom” and “loulou”. Note that “tom” is a probabilistic instance of Person with a probabilistic value P1 and it is a probabilistic instance of Animal with a probabilistic value P2 (same to “loulou”)
Probabilistic ObjectProperties	be-interested	This property represents a probabilistic relation between Person and Interest. It expresses that a person can be interested to one or more interests. For example, Jhon is interested to “music” with a probability of 0.5 and to “travel” with a probability of 0.2, etc
	be-connected	This property represents a probabilistic relation between the instances of class Interest. It expresses that an interest can be connected to one or more interests. For example, “watch-TV” and “Films” are connected with a probability of 0.9, etc
Probabilistic DataProperties	prefer-Animal	This DataProperty expresses that an person can prefer one or more animals. For example, John prefers cats with a probability of 0.5 and dog with a probability of 0.3, etc

describe the deterministic components necessary to satisfy the purpose of ontology (classes, proprieties, etc.). This step requires serious effort to analyze the domain of interest for identifying this knowledge. Table 1 resumes the probabilistic and classical components of O.

Construction of probabilistic ontology. After determining the domain and the purpose of ontology and identifying the different components of ontology (probabilistic and deterministic), the process of ontology development can be started by using a formal language such as OWL. For constructing a probabilistic ontology of a particular domain by using our proposed meta-model PODM (Hlel et al. 2016), firstly we create the probabilistic and deterministic concepts. The probabilistic classes of our example are represented as follows:

```
Declaration(ProbabilisticClass(a:Person))
Declaration(ProbabilisticClass(a:Animal))
```

Secondly, we create the probabilistic and deterministic properties of PO. The probabilistic proprieties of our example are represented as follows:

```
Declaration(ProbabilisticObjectProperty(a:be-connected))
Declaration(ProbabilisticObjectProperty(a:be-interested))
Declaration(ProbabilisticDataProperty(a:prefer-Animal)
```

The next step allows to create the deterministic and probabilistic instances of the ontology as well as the relations between them (populate the ontology further with instances). The probabilistic instances of our example as well as their proprieties are as follows:

```
//create the probabilistic individuals and its properties
Declaration(ProbabilisticNamedIndividual(a:loulou))
Declaration(ProbabilisticNamedIndividual(a:tom))
ProbabilisticClassAssertion(a:Person a:loulou 0,7^^xsd:float)
ProbabilisticClassAssertion(a:Person a:tom 0,3^^xsd:float)
ProbabilisticClassAssertion(a:Animal a:loulou 0,8^^xsd:float)
ProbabilisticClassAssertion(a:Animal a:tom "0,6^^xsd:float")
ProbabilisticPropertyAssertion(a:prefer-animal a:Smith “chat”
0,3^^xsd:float)) /* smith prefer chat with a probability with 0,3 */
//add the relations between the instances of ontology
ProbabilisticPropertyAssertion(a:be-interested a:tom a:imagery
0,5^^xsd:float))
ProbabilisticPropertyAssertion(a:be-connected a:visual-communication
a:imagery0,8^^xsd:float))
```

6 Conclusion and Perspectives

In this paper, we have presented how we can construct a probabilistic ontology for a particular domain by using our proposed meta-model of OWL2 (PODM). Our proposed method contains three phases which are Specification of requirements, Identification and description of certain (deterministic) knowledge and uncertain (probabilistic) knowledge and construction of PO. So, PODM can be used by users for creating probabilistic ontologies of complex domains.

In the future work, we will focus on the determination of probabilities which are associated to elements of ontologies for making them probabilistic. In addition, we will extend PODM with other probabilistic components: probabilistic axioms, probabilistic class expressions, etc.

References

Baader, F., Calvanese, D., McGuinness, D., Nardi, D., Patel-Schneider, P.F.: The Description Logic Handbook: Theory, Implementation and Applications. Cambridge University Press, New York (2003)

Ben Mrad, A., Delcroix, V., Piechowiak, S., Leicester, P., Abid, M.: An explication of uncertain evidence in Bayesian networks: likelihood evidence and probabilistic evidence-uncertain evidence in Bayesian networks. Appl. Intell. **43**(4), 802–824 (2015)

Costa, P.C.G., Laskey, K.B.: PR-OWL: a framework for probabilistic ontologies. In: FOIS, Amsterdam, pp. 237–249 (2006)

Ding, Z., Peng, Y.: A probabilistic extension to ontology language OWL. In: Proceedings HICSS, pp. 5–8 (2004)

Ding, Z.: BayesOWL: a probabilistic framework for uncertainty in semantic web. Thesis, Department of Computer Science and Electrical Engineering, University of Maryland, Baltimore, USA (2005)

Fabio, G., Rodrigo, B.P., Takiyamaa, F.I., Kate R.C.: Computing inferences for credal ALC terminologies. In: URSW, pp. 94–97 (2011)

Finn, V.J.: An Introduction to Bayesian Networks. UCL Press, London (1996)

Giugno, R., Lukasiewicz, T.: P-SHOQ(D): a probabilistic extension of SHOQ(D) for probabilistic ontologies in the semantic web. In: INFSYS, Austria, Research report (2002)

Gómez-Pérez, A., Fernández, L.M., Corcho, O.: Ontological Engineering with Examples from the Areas of Knowledge Management, e-Commerce and the Semantic Web. Springer, London (2006)

Hlel, E., Jamoussi, S., Turki, M., Ben Hamadou, A.: Probabilistic ontology definition meta-model. In: Czarnowski, I. et al. (eds.) Intelligent Decision Technologies. Smart Innovation, Systems and Technologies, vol. 56, pp. 243–254 (2016)

Hlel, E., Jamoussi, S., Hamadou, A.B.: A probabilistic ontology for the prediction of author's interests. In: Núñez, M., Nguyen, N.T., Camacho, D., Trawiński, B. (eds.) ICCCI 2015. LNCS (LNAI), vol. 9330, pp. 492–501. Springer, Heidelberg (2015). doi:10.1007/978-3-319-24306-1_48

Hlel, E., Jamoussi, S., Ben Hamadou, A.: Intégration d'un réseau bayésien dans une ontologie. In: IC 2014, pp. 295–297 (2014)

Horrocks, I., Sattler, U.: Ontology reasoning in the SHOQ(D) description logic. In: 17th Conference on Artificial Intelligence (IJCAI) (2001)

Laskey, K.B.: MEBN: a language for first-order bayesian knowledge bases. Artif. Intell. **172**(2–3), 140–178 (2008)

Motik, B., Patel-Schneider, P.F., Parsia, B.: OWL2 Web Ontology Language: Structural Specification and Functional-Style Syntax (2012)

Stephen, K.R., Adam, P.: A framework for constructing cognition ontologies using WordNet, FrameNet, and SUMO. J. Cogn. Syst. Res. **33**, 122–144 (2015)

Salvatore, F.P.: An individual-centric probabilistic extension for OWL: modeling the uncertainness. In: International Conference on Computational Science, pp. 1742–1751 (2015)

Schmidt-Schauss, M., Smolka, G.: Attributive concept descriptions with complements. Artif. Intell. **48**(1), 1–26 (1991)

Predoiu, L., Stuckenschmidt, H.: Probabilistic models for the semantic web: survey. In: Web Technologies: Concepts, Methodologies, Tools, and Applications, pp. 1896–1928 (2010)

Yang, Y.: A framework for decision support systems adapted to uncertain knowledge. Thesis, der Universitat Fridericiana zu Karlsruhe (TH) (2007)

Yang, Y., Calmet, J.: Ontobayes: an ontology-driven uncertainty model. In: IAWTIC 2005, pp. 457–464 (2005)

A HMM-Based Arabic/Latin Handwritten/Printed Identification System

Ahmed Cheikh Rouhou[1(✉)], Zeineb Abdelhedi[1], and Yousri Kessentini[1,2]

[1] MIRACL Laboratory, Digital Research Center of Sfax, University of Sfax, Sfax, Tunisia
cheikhrouhouahmad@gmail.com, zeineb.abdelhedii@gmail.com
[2] LITIS Laboratory EA 4108, St Etienne du Rouvray, France
yousri.kessentini@litislab.eu

Abstract. For document analysis and recognition systems, script identification is considered as an important preprocessing step in the design of multi-scripts OCR system. In this paper, we propose a novel HMM based identification system to recognize on only one level the writing type (handwritten or machine-printed) and the script nature (Arabic or Latin) of the input image. The proposed system is based on Histogram of Oriented Gradient (HOG) features which have demonstrated an interesting properties for script characterization. Experiments have been conducted on word and line images collected from public databases and show promising results.

1 Introduction

In document analysis and recognition research field, statistics confirm that Latin script is mostly used. With the increasing communication between different word communities, more scripts are getting integrated. Recently, various recognition systems have been developing to recognize multi-scripts documents where more than a script is presented. Yet, it is difficult to propose a unified system covering various scripts. Each script has its proprieties and specific processing techniques. Recognizing multi-script document images can be achieved by placing an automatic preprocessing module. Such module includes algorithms that identify scripts and decide the suitable recognition engine to use further. Recognizing documents' script, called script identification in the document analysis community, is not a simple task. In fact, many scripts have similar visual representations especially handwritten scripts due to their cursive nature. Besides, some printed text fonts are similar to handwritten text presentations.

Among the relevant research contributions, different approaches are applied to identify Arabic script. The presented works in [1,8] aim to identify Arabic and Latin scripts in text blocks based on K-Nearest Neighbor (KNN) and a set of global features. In [4], authors used the KNN to identify Farsi and Arabic scripts

A. Abraham et al. (eds.), *Proceedings of the 16th International Conference on Hybrid Intelligent Systems (HIS 2016)*, Advances in Intelligent Systems and Computing 552,
DOI 10.1007/978-3-319-52941-7_30

for word images by utilizing structural features, histogram features and profile features. Moreover, Support Vector Machine-based system was proposed by [9]. This system is capable of identifying up to 11 scripts including Arabic using SVM classification and Gabor extracted features. Although, different works are published for other scripts: [2,5,7] systems' identify Latin, Devanagari or Bangla scripts in text line images. Then, [3,6] presents prototypes able to detect Latin scripts using word level representation. Lately, Saidani et al. [20] published an Arabic/Latin and machine-printed/handwritten word discrimination approach based on HOG shape descriptor. They improved regular HOG-based features for script and nature identification. The discrimination is done using a Bayes-based classifier and gave excellent results. A complete survey on other script identification works is given in [19].

Besides, Hidden Markov Model-based contributions are very limited. At first, authors in [10] proposed an identification system that separates handwritten material from machine printed Latin text using HMM. The proposed approach can only be applied to English or other Latin languages due to the nature of the HMM defined in the approach. Next, in [11], authors proposed a system able to identify 18 scripts corresponding to 54 languages for printed text images. This system is very complex due to the huge number of scripts involved. Authors tackled this issue by proposing two solutions: at first, authors have proposed a solution based on a single ergodic HMM model collecting characters of all scripts called "union OCR model". With up to 24000 models, this prototype is impractical to be integrated in OCR systems. After that, the previous models are minimized to 25 script models named "Script-HMM" where each model denotes a script.

This work extends the existing script identification approach with the following benefits:

- We propose a one level HMM-based identification system to identify the script nature and the writing type (handwritten or machine-printed) contrary to [8] where three decision levels are needed.
- The natural writing direction (left/right for Latin and right/left for Arabic) is considered as a meaningful information to characterize the script nature thanks to the use of HMM which can efficiently integrate the contextual information.
- The proposed approach can operate on line level to avoid the word segmentation problems like the inconsistent spacing between words and PAWs in the case of Arabic script.

The remainder of the paper is organized as follows. We present the differences and similarities of Arabic and Latin scripts in Sect. 2. Afterwards, we describe our HMM-based script identification system. Later on, we introduce experimental results in Sect. 4. Finally, we state the conclusions in the last section followed by some perspectives.

2 Characteristics of Arabic and Latin Scripts

The Latin script is different from Arabic script on certain aspects such as the natural direction of the writing and the alphabet. It is written from left to right as opposite to the Arabic that is from right to left.

The Latin script alphabet contains 26 characters. Words are presented as succession of characters in two styles: chained (see Fig. 1a) or separated (see Fig. 1b). Further, each letter has capital and lowercase format (see Fig. 1b).

Arabic script has more complexities than Latin. First of all, it contains 28 alphabetic letters and each one has a different shape according to its position (up to 124 different shapes totally) within a word. Next, Arabic words are made with sequences of chained letters. In fact, these sequences define connected components called Part of Arabic Words (PAWs), as Fig. 1d illustrates. In addition, diacritic are often used in letters and their presence distinguishes characters from each other. To end with, ligatures, vertically chained letters, could be found in handwritten and printed Arabic texts (as indicated in Fig. 1c).

In spite of these differences between the Arabic and Latin, there are some similarities and common writing properties. Both scripts have writing lines (baseline, mean-line ...) with a central band where pixels are concentrated. Also, variability between different writers results into inconsistent shapes for each letter. Figure 1d shows a letter differently presented by a single writer. Given these characteristics, recognizing Arabic and Latin scripts is a complex and challenging task. Accordingly, we had chosen to use discriminating features based on Histogram of Oriented Gradient [18]. This descriptor is efficient in terms of differentiating between handwritten/machine-printed, Arabic and Latin scripts. Also, we had used Hidden Markov Model (HMM) to model scripts for each document nature. HMMs have a huge ability to integrate writing direction and absorb writers variability.

Fig. 1. Samples from each script class. (a) handwritten Latin word. (b) machine-printed Latin word presenting the letter 'R' in capital and lowercase 'r' format. (c) Arabic printed word image with vertical ligatures. (d) handwritten Arabic word containing 9 PAWs.

3 Arabic-Latin Nature/Script Identification System

The proposed HMM-based identification system discriminates Arabic and Latin scripts for both writing types: handwritten and machine-printed line images. In the first step, preprocessing is applied to the input image. The next step is the features extraction where a sliding window is shifted along the input image in two direction from right to left and from left to right to conserve the natural

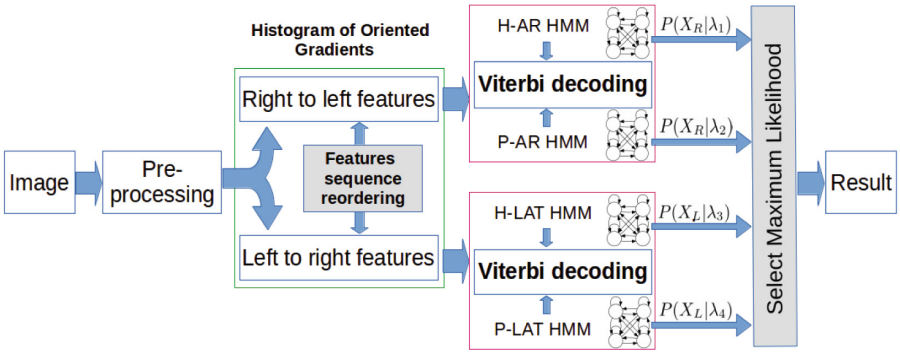


Fig. 2. System overview

writing direction of Arabic and Latin scripts. Avoiding time consumption, after producing a features sequence for a direction, the other one is automatically generated by inverting the order of features vector within the sequence. To model the script nature and the writing type, we define the following 4 HMMs: Handwritten Arabic HMM (H-AR HMM), Printed Arabic HMM (P-AR HMM), Handwritten Latin HMM (H-LAT HMM) and Printed Latin HMM (P-LAT HMM). These HMMs are learnt from the script specific training corpus as specified in Sect. 3.3. The identification of the script nature and the writing type is done by processing each input image using Viterbi algorithm and finding the maximum probability path between the 4 defined HMMs. In this step, features extracted form right to left are decoded using only the 2 models P-AR and H-AR HMMs, and features extracted form left to right are decoded using the 2 models P-LAT and H-LAT HMMs as shown in Fig. 2.

3.1 Preprocessing

Word and line images are collected from public databases. These samples are extracted from scanned documents. Mostly, the scanning process returns noisy and distorted images. To maintain stable performances for our system, three preprocessing tasks are applied for collected images.

- **Normalization:** Line images may contain rotated images and slanted content. These facts confuse the character modeling process and leads to erroneous returns. To normalize images, Deslant and Deskew algorithms are applied to correct these problems.
- **Median Filter:** noisy pixels are often detected in document images due to scanning process (old documents, scanner quality...). Therefore, Median filter is applied to suppress these unwanted entries without losing meaningful data. This filter is a smoothing technique that preserves characters edges in text images.

– **Padding removal (Cropping):** applying normalization tasks, filters and segmentation on document images results in meaningful images with extra empty regions (white pixels). These regions do not affect system output and mechanism but consumes useless iteration in features extractions. Consequently, paddings are removed to increase precision and response time.

3.2 Feature Extraction

Features need to represent relevant properties of the scripts: On the one hand, the printed script is mono-spaced, has a uniform dimension and has a stable base-line. On the other hand, especially for handwritten text, there is no formality in writing style and each character could have many presentations. Hence, we use Histogram of Oriented Gradient (HOG) [18] as feature descriptor for the discrimination of handwritten or machine-printed Arabic and Latin scripts. Indeed, in [20], authors made a detailed study of HOG descriptor application and proved its efficiency in identifying Arabic/Latin script and its nature.

The main idea of HOG is representing the different shapes and objects. The latters are described by the distribution of local intensity gradients or edge detections. The method implementation follows a sequence of tasks. During the HOG process, the image is divided into small regions which called "cells". The histogram of gradient directions or edge orientations are calculated for the pixels in each cell. Then, the histograms are combined for the representation of the descriptor. To ameliorate the result, the local histograms are normalized by using contrast. It is done by collecting a measure of intensity gradients per block, which covers a larger area than a cell. Following the cells within a block are normalized according to the calculated block measure. The purpose of this improvement is attaining the better invariance to illumination and shadowing.

In order to build the feature vector sequence, the image is divided into vertical overlapping windows or frames. The sliding window is shifted along the word image from right to left (case of Arabic words) or left to right (case of Latin words) and HOG features are computed for each frame. These features use a zoning approach and extract a set of features that is based on the gradient orientation. The dimensionality of the feature vectors used in our experiments is 64.

We have mentioned the mechanism of HOG features extraction and how we deployed this descriptor in our work. In this sense we explain the major factors that made the HOG features efficient for our system. Obviously, the obtained features in different cells imitate the morphology of the script. The written material for the scripts in the both natures are visually composed of three parts: the upper zone (two upper adjacent cells), the middle zone (four middle cells) and the lower zone (two lower cells). The middle zone presents the writing central band for both scripts. It contains the highest pixels distribution in the images. Moreover, this zone denotes the common properties of the two scripts. To emphasize, the printed script has a high gradients orientation value on the vertical/horizontal axes, unlike the handwritten script which has high orientation values on the diagonal axes. In contrast, the upper and lower window cells make the difference between printed scripts. Arabic is easily detected

by the produced HOG values of the slanted descending characters and diacritics. Meanwhile, spaces between printed Latin script characters reduce the HOG values. For handwritten texts, the HOG features conserve the natural direction of Arabic and Latin scripts based on the gradient orientation. In case of Arabic, the diagonal stroke orientations are generally from right to left as opposite to the Latin, that is from left to right.

The main idea of our script identification system is the discrimination between Arabic and Latin scripts for both Handwritten and machine-printed writing types. In order to do that, we have chosen to use the Hidden Markov Models (HMM) given their abilities to learn a better representation of the script specificities and to absorb the writing variability. Respectively, we define four basic HMM models λ_i for each nature/script: λ_1 for Handwritten Arabic, λ_2 for Printed Arabic, λ_3 for Handwritten Latin and λ_4 for Printed Latin. All models are defined by an ergodic structure concatenating all character models of the corresponding script.

Our system takes as input the word/line image represented by their corresponding features sequences and the four ergodic HMMs modeling the different writing natures per script type. Given that Arabic and Latin scripts have a different natural writing direction, the decoding is processed in two steps. Denoting $X_R = x_R^1, x_R^2, ..., x_R^n$ as the features' sequence extracted from right to left, the first decoding step uses Viterbi algorithm to decode X_R using the two Arabic models; λ_1 (Handwritten Arabic HMM) and λ_2 (Printed Arabic HMM). This first decoding step returns two likelihood: $P(X_R|\lambda_1)$ and $P(X_R|\lambda_2)$.

In the second step, the same decoding process as above is repeated, using the features' sequence $X_L = x_L^1, x_L^2, ..., x_L^n$ extracted from left to right and the two Latin script models; λ_3 (Handwritten Latin HMM) and λ_4 (Printed Latin HMM), are used for the decoding step. The Viterbi algorithm returns two more likelihood; $P(X_L|\lambda_3)$ and $P(X_L|\lambda_4)$.

Finally, the script nature and the writing type prediction is done by finding the maximum likelihood as described in Eq. 1:

$$\lambda^* = argmax_{\lambda_i}(P(X_R|\lambda_1), P(X_R|\lambda_2), P(X_L|\lambda_3), P(X_L|\lambda_4)) \tag{1}$$

3.3 Models Training

In order to model the Latin characters, we built 72 models corresponding to the 52 upper and lower case Latin characters, numerical digits and accented letters. In the case of Arabic characters, we built up to 150 character models. An Arabic character may actually have different shapes according to its position within the word (beginning, middle, end word position). Other models are specified with additional marks such as "shadda". In both Latin and Arabic script, each character model is composed of four emitting states. The observation probabilities are modeled with Gaussian Mixtures. Embedded training is used where all character models are trained in parallel using Baum-Welch algorithm applied on word examples.

4 Experiments and Results

Three sets of datasets are prepared to test our system depending on the structure of the input data: word level, line level and document level.

4.1 Word Level Results

To evaluate the performance of our identification system, a data corpus is collected from different public word datasets. For the Arabic script, handwritten samples (HA) are collected from the IFN/ENIT [12] database containing handwritten word images written by 411 writers. The printed dataset is collected from the APTI database [13]: 6 fonts (Andalus, Arabic Transparent, DecoType Naskh, Tahoma, Simplified Arabic and M Unicode Sara), and 6 font sizes (6, 8, 10, 12, 14 and 16) are used. Latin handwritten word images are extracted from the RIMES database [17]. Printed word images are collected from the newly published database ALTID [14] which is under development and we manually extracted word images from Latin printed text blocks. For each script and each writing type, 4500 word images are collected divided into 3000 for training and 1500 for the test. In total, the database is composed of 12000 word images for training and 6000 word images for test.

A detailed analysis is presented in Table 1 presenting confusion matrix. The matrix lines represent the true value of the script type/writing nature and columns represent the output identification rates of our system. The mean identification rate is 91.55%. The analysis of the results shows that the proposed system identifies in a reliable way Latin printed and Arabic handwritten scripts with a perfect accuracy. In addition, there is some confusion between the Handwritten and printed Arabic. These error rates are due to the use of multi-font/multi-size printed words that cover various complexities of shapes.

We notice that Latin confused samples correspond generally to short words (maximum 4 characters per word), because less information are extracted from the image. Arabic handwritten and printed confusions are due to the variability of handwriting and the use of multi-font/multi-size printed words. This can also be explained by the bad quality of some printed Arabic word images.

Table 1. The word-level script identification system performances

	H-AR	P-AR	H-LAT	P-LAT
H-AR	**98.34**%	1.64%	0.2%	0%
P-AR	9.86%	**90.14**%	0%	0%
H-LAT	18%	4%	**78**%	0%
P-LAT	0.13%	0%	0.13%	**99.74**%

4.2 Line Level Results

Automatic word segmentation stills a hard task especially for the handwritten document images. Obviously, lines could be easily detected within a document images for both scripts. We evaluate our system on line level using a collection of 16000 line images: 3000 images of writing natures per script type for the model training and 1000 for the test. Latin text line images are gathered from the same word level datasets [14,17]. But, the Arabic handwritten texts are obtained from the KHATT [15] wish contains 9327 line images written by 1000 persons. The printed text lines are extracted from the KAFD database [16] which contains document images printed in 40 fonts and in 10 sizes. In addition, we segmented document images of [14,17] into line images using the Tesseract OCR engine [22].

Table 2. The line-level script identification system performances

	H-AR	P-AR	H-LAT	P-LAT
H-AR	**99.8**%	0.1%	0%	0.1%
P-AR	0.1%	**90.9**%	9%	0%
H-LAT	11%	0%	**89**%	0%
P-LAT	0%	0.1%	0.3%	**99.6**%

Table 2 presents the different identification and confusion rates. With 94.82% mean identification rate, the script identification results have been enhanced by the use of line level images with a gain of 3.27% compared to the word level. In fact, line images contain more information compared with word images which influence positively on the decision made by the decoding step.
A large number of the confused samples contain large area filled with noisy pixel. These samples are the result of segmentation errors. Manual selection of the test samples enhanced the results of printed Arabic and handwritten Latin to reach respectively 97.8% and 91.2%. The other confusions are made due the same images conditions cited for the word level images.

4.3 Document Level Results

To evaluate our system on document level, text lines are first segmented from the document images using the method presented in [21]. Using the same datasets, we collected 182 handwritten document images for the handwritten Latin, 246 documents for the handwritten Arabic, 115 for the printed Arabic and finally 234 for the printed Latin.

In the next step, each line image is processed by our identification system to predict its script nature and its writing type. Given the decision of its line images, the document script identification is done by selecting the maximum score of its corresponding lines. Clearly, the maximum score is given to the line image that contains high number of characters. Document level script identification results

Table 3. The document-level script identification system performances

	H-AR	P-AR	H-LAT	P-LAT
H-AR	**100**%	0%	0%	0%
P-AR	0%	**99.2**%	0.8%	0%
H-LAT	6.9%	0%	**93.1**%	0%
P-LAT	0%	0%	0%	**100**%

are good as it is shown in Table 3. The mean of the identification rates has been increased to reach 98%. We believe also that a robust segmentation algorithm will enhance these results for any document introduced to the system.

5 Conclusion

This paper presents a novel script identification system based on HMM. The originality of our approach consists in discriminating in only one level, handwritten or machine-printed, Arabic and Latin scripts. To do so, features are extracted in two directions (left-to-right and right-to-left) to support scripts' properties. The identification of script type and writing nature is then performed by applying a particular Viterbi decoding that uses four ergodic HMMs: the model that yields the highest likelihood is then selected as the recognized class. For experimental evaluation, the developed system was tested using some parts of public databases. The obtained results are fairly encouraging. Future works consist in generalizing our system to deal with more scripts. Finally, we aim to reduce the computational complexity of our identification system by decreasing the number of states in the ergodic HMMs using a state HMM clustering algorithm.

References

1. Baati, K., Kanoun, S., Benjlaiel, M.: Diffirenciation d'ecriture Arabe et Latine de natures Imprimee et Manuscrite par approche globale. In: Proceedings of Colloque International Francophone sur l'ecrit et le Document CIFED, pp. 313–324 (2010)
2. Kavallieratou, E., Stamatatos, S.: Discrimination of machine-printed from handwritten text using simple structural characteristics. In: Proceedings of the 17th International Conference on Pattern Recognition, ICPR 2004, vol. 1, pp. 437–440, 23–26 August 2004
3. Zhou, L., Lu, Y., Tan, C.L.: Bangla/English script identification based on analysis of connected component profiles. In: Bunke, H., Spitz, A.L. (eds.) DAS 2006. LNCS, vol. 3872, pp. 243–254. Springer, Heidelberg (2006). doi:10.1007/11669487_22
4. Mozaffari, S., Bahar, P.: Farsi/Arabic handwritten from machine-printed words discrimination. In: 2012 International Conference on Frontiers in Handwriting Recognition (ICFHR), pp. 698–703, 18–20 September 2012
5. Pal, U., Chaudhuri, B.B.: Script line separation from Indian multi-script documents. In: Proceedings of the Fifth International Conference on Document Analysis and Recognition, ICDAR 1999, pp. 406–409, 20–22 September 1999

6. Faria da Silva, L., Conici, A., Sanchez, A.: Automatic discrimination between printed and handwrittentext in documents. In: 2009 XXII Brazilian Symposium on Computer Graphics and Image Processing (SIBGRAPI), pp. 261–267, 11–15 October 2009
7. Pal, U., Chaudhuri, B.B.: Machine-printed and hand-written text lines identification. Pattern Recogn. Lett. **22**(3–4), 431–441 (2001)
8. Benjelil, M., Kanoun, S., Alimi, A.M., Mullot, R.: Three decision levels strategy for Arabic and Latin texts differentiation in printed and handwritten natures. In: Ninth International Conference on Document Analysis and Recognition, ICDAR 2007, vol. 2, pp. 1103–1107, 23–26 September 2007
9. Pati, P.B., Ramakrishnan, A.G.: Word level multi-script identification. Pattern Recogn. Lett. **29**(9), 1218–1229 (2008)
10. Guo, J.K., Ma, M.Y.: Separating handwritten material from machine printed text using hidden Markov models. In: Proceedings of the Sixth International Conference on Document Analysis and Recognition, pp. 439–443 (2001)
11. Genzel, D., Popat, A.C., Teunen, R., Fujii, Y.: HMM-based script identification for OCR. In: Proceedings of the 4th International Workshop on Multilingual OCR, article 2. ACM, New York (2013)
12. El Abed, H., Margner, V.: The IFN/ENIT-database - a tool to develop Arabic handwriting recognition systems. In: 9th International Symposium on Signal Processing and Its Applications, ISSPA 2007, pp. 1–4, 12–15 February 2007
13. Slimane, F., Ingold, R., Kanoun, S., Alimi, A.M., Hennebert, J.: A new Arabic printed text image database and evaluation protocols. In: 10th International Conference on Document Analysis and Recognition, ICDAR 2009, pp. 946–950, 26–29 July 2009
14. Chtourou, I., Cheikh Rouhou, A., Kallem, F., Kanoun, S.: ALTID: Arabic/Latin text images database for recognition research. In: 13th International Conference on Document Analysis and Recognition, ICDAR 2015 (2015)
15. Mahmoud, S.A., Ahmad, I., Al-Khatib, W.G., Alshayeb, M., Parvez, M.T., Margner, V.: KHATT: an open Arabic offline handwritten text database. Pattern Recogn. PR **47**(3), 1096–1112 (2014)
16. Hamzah, L., Mahmoud, S.A., Sameh, A.: KAFD Arabic font database. Pattern Recogn. PR **47**(6), 2231–2240 (2014)
17. Grosicki, E., El Abed, H.: ICDAR 2009 handwriting recognition competition. In: 10th International Conference on Document Analysis and Recognition, ICDAR 2009, pp. 1398–1402, 26–29 July 2009
18. Rodriguez, J., Perronnin, F.: Local gradient histogram features for word spotting in unconstrained handwritten documents. In: Proceedings of International Conference on Frontiers in Handwriting Recognition (ICFHR 2008), pp. 7–12 (2008)
19. Ghosh, D., Dube, T., Shivaprasad, A.P.: Script recognition: a review. IEEE Trans. Pattern Anal. Mach. Intell. **32**, 2142–2161 (2009)
20. Saidani, A., Kacem, A., Belaid, A.: Arabic/Latin and machine-printed/handwritten word discrimination using HOG-based shape descriptor. ELCVIA Electron. Lett. Comput. Vis. Image Anal. **14**, 1–23 (2015)
21. Smith, R.W.: Hybrid page layout analysis via tab-stop detection. In: 10th International Conference on Document Analysis and Recognition, pp. 241–245 (2009)
22. Smith, R.: An overview of the Tesseract OCR engine. In: Proceedings of the Ninth International Conference on Document Analysis and Recognition, ICDAR 2007, vol. 02, pp. 629–633 (2007)

Vision Based Hand Gesture Recognition for Mobile Devices: A Review

Houssem Lahiani[1,3,4(✉)], Monji Kherallah[2], and Mahmoud Neji[3,4]

[1] National School of Electronics and Telecommunications, University of Sfax, Sfax, Tunisia
lahianihoussem@gmail.com
[2] Faculty of Sciences, University of Sfax, Sfax, Tunisia
monji.kherallah@gmail.com
[3] Faculty of Economics and Management, University of Sfax, Sfax, Tunisia
mahmoud.neji@gmail.com
[4] Multimedia Information Systems and Advanced Computing Laboratory, Sfax, Tunisia

Abstract. The desire to interact with a mobile device in an intuitive and natural way is growing. In fact, research in this field aims to develop systems able to model, analyze and recognize user's hand gestures to control mobiles without having the need to touch the screen. We give in this paper an overview of current research works and an analysis of comparative studies in this field. This paper focuses on the main steps of hand gesture recognition for mobile devices like detection, tracking and recognition. This work also gives an analysis of the existing literature on gesture recognition systems for human-computer interaction by classifying them under various key parameters. At the end we conclude with some reflections on future works.

Keywords: Hand gesture · Recognition · Mobile device · Human computer interaction · Android

1 Introduction

In these times, the interaction with a mobile device has become greater than ever and it has become a necessity and we couldn't live without it. This embedded technology has become so integrated in our daily life and we became obliged to use it in almost all areas of our lives. To effectively exploit those devices, we need more and more interaction with them. An interaction is a manipulation of graphic objects like icons and windows using a touchscreen or a pointing device. Even if the invention of virtual keyboard and the touchscreen represents a great progress, there are some situations in which these pointing devices are incompatible for Human-Computer Interaction. The use of hand gestures offers an interesting alternative to these cumbersome devices for human-computer interaction. Using the hand as a device can help to communicate with mobile devices in a more natural and intuitive way. Hand gestures are therefore a non-verbal means of communication. Hand Gestures can be static (pose or posture) which need less computational complexity or dynamic which could be more complex.

A. Abraham et al. (eds.), *Proceedings of the 16th International Conference on Hybrid Intelligent Systems (HIS 2016)*, Advances in Intelligent Systems and Computing 552,
DOI 10.1007/978-3-319-52941-7_31

To recognize gestures, some gesture recognition methods, in addition to the camera, used additional hardware such as gloves and sensor to easily extract the full description of the gesture features. But there are other methods like appearance based methods that use the skin color to segment hand and then extract features. These methods are more easy, intuitive, natural and cost less compared to methods mentioned previously. The remainder of this paper is organized as follows:

Section 2 gives a literature review and explains key issues of hand gesture recognition system for mobile devices. Section 3 discusses application areas of hand gesture recognition systems for mobile devices. In Sect. 4 a summary of recent research results is shown and a possible future work is given, and finally we conclude in Sect. 5.

2 Literature Review and Key Issues of Hand Gesture Recognition for Mobile Devices

The implementation of an efficient hand gesture recognition system for mobile devices is aided through two main kinds of enabling technologies for human-machine interaction such as contact based and vision based devices. Contact based devices require physical interaction like accelerometers, data glove and multi-touch screen and they do not provide much of acceptability because their use is uncomfortable especially for an inexperienced user [1]. Thus vision based approaches have been employed for hand gesture recognition in human-computer interaction. Contact-based approach for hand gesture recognition is so embarrassing, hence vision-based approach is a comfortable experience but it is hard to deploy it in bad conditions. The main challenge of the recognition of hand gestures by a mobile device based on vision based approach is to cope with the wide variety of gestures in addition of the computational limitations of mobile devices. Note that the location of the hand in this context is a computer vision problem and hand gesture recognition is a machine learning problem which makes the fact to develop a real-time gesture recognition system by mobile device, which has a limited capacity in terms of CPU, memory and battery, challenging. Most researchers classified hand gesture recognition systems mainly in four steps after image acquisition from camera or data glove instrumented device. The captured hand gestures by the camera of the device constitute the input of the system. Systems have to work with approaches that are computationally inexpensive to compensate the weak processing capability of mobile device. These steps are mainly: hand detection, hand segmentation, feature extraction and gesture classification as illustrated in Fig. 1.

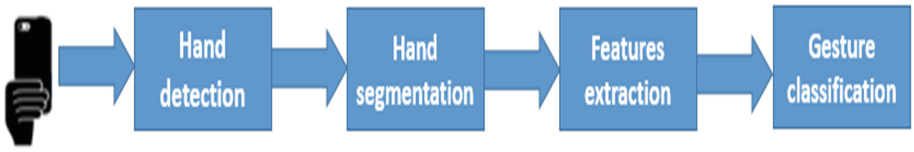


Fig. 1. Block diagram of gesture recognition system

Image acquisition is the first step in gesture recognition system. Frames are captured by the smartphone camera then the hand detection is processed. Then the segmentation of hand is done. After that hand tracking and segmentation are done, fetures must be extracted from the segmented hand. And finally, the classification step is processed. In this step input features are compared with features' of the trained database.

2.1 Hand Detection and Image Pre-processing

For any vision-based system, Image Acquisition is the step only after which we can go forward with image processing then the detection step can be processed and we can go forward to segmentation of the hand to subtract the region of interest from the background. This segmentation is crucial because it isolates the relevant area from the rest of background image before transmitting it to tracking and recognition steps. Also the moving object could be extracted from the background using a threshold [4]. Several methods have been proposed in the literature using many types of visual features and, in several cases, their combination. Among the most used cues in hand gesture recognition systems for mobile device we find the color and the shape of the hand. The cue commonly used in gesture recognition for mobile devices, to segment the hand, is the color of the skin, because it is easy and invariant to translation and rotation changes [2] and because it is very suitable for embedded systems like mobile devices since color is computationally inexpensive, and it can give more information than a luminance-only image or an edge-segmented image which need more computational resources which make real-time systems hard to realize [3]. However some factors can obstacle segmentation process like illumination changes, complex background and low video quality. Therefore, to enhance the segmented image, there are preprocessing operations which can be applied such as subtraction and normalization. Indeed, the shape of hand can utilized to detect it in images in several ways. Several information could be obtained by the contour extraction (Edge detection) of the desired object (hand). If it is detected correctly, the hand's shape can be presented by this contour and did not dependent on skin color and illumination.

H. Lahiani et al. in [5, 6] makes a real time hand gesture recognition system for android devices based on skin color segmentation. L*A*B color space was used because it makes separation between chrominance components and illumination.

In [7] V. ShirshirReddy et al. developed a system based on finger detection to control tablets. The finger is detected by putting an orange color round mark of known size on the front side of the finger of user. The round shape is chosen since it is simple to detect and the choice of the orange color is done because it is easy to differentiate it from surroundings in most of cases.

In [8] Prasuhn make a static hand gesture recognition system for the American Sign Language. To extract the hand area he applies simple binarization in HUV color space. Then binarized image has been submitted to morphological operations to reduce noise. Specifically, he applies closing first and then opening operation [9].

To recognize the shape of the hand, A. Saxena et al. [10] and J.L. Raheja et al. [11] used Sobel Edge Detector for image pre-processing for a sign recognition system using a mobile device.

To make a real time Indian Sign Language Interpreter for mobile devices S. Swamy et al. [12] used the Viola-Jones algorithm based on LBP features to recognize the posture of the hand. For this system, the pure background color is required during the image capture progress. To tackle the problem of lighting condition and background noise HSV model's threshold value was used to separate the background noise in the training images.

To make a Sign Language system based on android device for deaf people Setiawardhana et al. [13] used the Viola-Jones algorithm to detect the hand of the user and then to recognize finger alphabet. For this system, the detection of skin color, noise removal and thresholding are processed after capturing the image.

To make a Vision-based Gesture Recognition and 3D Gaming System for Android Devices controlled by hand gestures, Mahesh B. Mariappan et al. [14] used Cascaded Haar Classifiers to efficiently track object (hand) by the mobile device. After getting the image from the live video the image is converted to a grayscale image using the cvtcolor function. Then the contrast of the image is improved because Haar classifiers use Haar features for classification purposes and those features are contrast based rectangular features.

In [16] to make a Static Hand Gesture Recognition using an Android Device, Tejashri J. Joshi et al. used thresholding to segment hand from background. They replace each pixel in the image by black pixel when the image intensity is less than threshold and a white pixel when the image intensity is greater than threshold. Then a rotation, cropping and normalization are done to the binary image.

2.2 Features Extraction

Good feature extraction is conditioned by a good segmentation and which have an important role in a good recognition process. The segmented hand image features vector can be extracted in several ways. Many methods were used to represent how features could be extracted. Some methods have used the shape of the hand, like the contour and the silhouette of the hand and some others use palm center, fingertips position, etc. For the features extraction step, in [5, 6, 19] H. Lahiani et al. used OpenCV functions to return the contour of the hand and convex hull points.

OpenCV have functions which return the contour and convex hull points of the hand from the segmented image. Using the bounding box with those elements they arrived to calculate the radius of inscribed circle of contour and the center of the palm. After that, location of Fingertips are computed using defect points and finally finger vectors are calculated and divided by the radius of the circle to obtain final feature vector.

In [8] to represent hand shape, L. Prasuhn et al. used the descriptor of Histogram of Oriented Gradients (HOG). HOG feature is robust under illumination change but its performance is vulnerable under object rotation. Although it is suitable for hand pose estimation. HOG can give a feature vector when an input image of fixed size is given. The proposed system used parameters of the original HOG paper [15].

In [12] S. Swamy et al. used LBP features. In this work LBP descriptors are experimented and they show a discriminative power in the classification of hand shape and their capability to capture enough information in order to distinguish among hand shapes in the classification phase. In [16] Tejashri J. Joshi et al. used Principal Component Analysis (PCA) in the features extraction step. At first a set "S" of "m" training images was created and each image was transferred into vector of size 1*5400. Then a mean image was computed and the difference between each training image and the mean image was calculated to obtain the covariance matrix. Then Eigen vectors of covariance matrix was computed and finally the image was projected into space using feature of each image in the training dataset. Feature vector of all training images are computed and those feature vectors was utilized to train a classifier.

In [10] A. Saxena et al. used hand token as feature to make the image into a usable form for neuronal network. The cosinus and sinus angles of the shape where used to represent the criteria of a recognition pattern.

In [7] V. Shirshir Reddy et al. to track the finger which controls the tablet they extracted the area and centroid of the orange mark placed in the finger whose position is traced at any time with respect to an initial position named "O".

Setiawardhana et al. [13] used the Viola-Jones Algorithm to detect the hand of the user and Mahesh B. Mariappan et al. [14] used Cascaded Haar Classifiers to efficiently track hand (closed fist) so as features they used Haar features. Features proposed by Viola and Jones consider detection windows delimiting adjacent rectangular areas; pixels intensities of these blocks are added, forming sums whose difference is a feature. A feature therefore is a real number which codes the pixel-wise variations of content at a given position in the detection window. The presence of edges or changes in texture are thus digitally translated by the values of Haar features.

Setiawardhana et al. [13] after detecting the hand and the preprocessing step, the hand shape image is obtained as input data and training data.

2.3 Gesture Classification and Recognition

The ultimate objective of hand gesture recognition is to interpret the semantics that the hand posture or gesture want to convey. After modeling and analyzing the image of the hand, a gesture classification method is needed to recognize gesture. The purpose of this process is usually to classify captured movements into different types of actions. Figure 2 explain the architecture of the classification system.

Two types of gesture recognition can be distinguished: the static or dynamic gesture recognition, based on one or more frames respectively. Most of the vision based systems done to recognize hand gestures by a mobile device are destined for static hand gesture recognition. Static hand gestures are classified into linear and non-linear learner. The first one is adapted to the linearly separable data and the second for the other cases. Another way to classify learning algorithms is to consider their results. Thus, it distinguishes supervised learning, unsupervised learning, semi-supervised learning, etc. The choice of the training algorithm depends mainly on the selected hand gesture representation. For example, S. Swamy et al. [12] proposed to recognize static hand gesture have choose LBP feature to apply The Adaptive Boosting Learning Algorithm

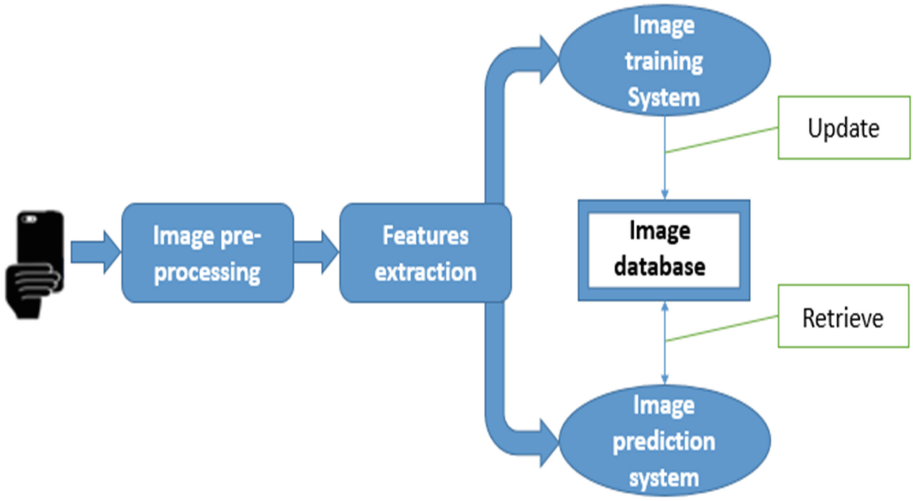


Fig. 2. Architecture of the classification system

"AdaBoost" which represents a learning algorithm that is capable of integrating the information of a category of objects. This algorithm is used by Viola-Jones algorithm to train samples' set which includes cascade based classifier. AdaBoost combines weak classifiers that cannot give satisfactory results to become a strong classifier to obtain the better result. The Adaptive Boosting learning algorithm gets the best weak classifier from an images' set that contains positive and negative images. After selecting the best weak classifier, Adaboost algorithm adjust the weights of the training images. At this stage, weights of classified training images decreased and unclassified images increased. After that, unclassified images are more focused by the Adaboost algorithm which tries to classify correctly the misclassified images. To recognize finger alphabets Setiawardhana et al. [13] after detecting the hand of the user by using the Viola-Jones Algorithm which use AdaBoost Algorithm cited above, they used K-nearest neighbor (k-NN) algorithm as a classifier to build the classification model which is a non-parametric lazy (no explicit training phase or it's minimal) learning algorithm. Each data point in the set of data is considered in a known class. Then, the classification of a new data point can be predicted based on the known classifications of the observations in the database. Indeed, the database is known as the training set. The classification of a new observation is based on the classifications of the observations in the database which it's 'most similar' to. Neighbors are usually calculated using Euclidean distance.

H. Lahiani et al. in [5, 6] proposed to recognize static hand gesture used Multi-class SVM (Support Vector Machine) to build the training classification model and to make predictions. SVMs are a set of supervised learning techniques aimed at tackling discrimination and regression problems. SVM are a generalization of linear classifiers. SVM can be used to solve discrimination problems, in other meaning it decides to which class a sample belongs, or regression problem, in other meaning it predicts the numerical value of a variable. The resolution of these two problems is through the construction of a function h to which an input vector x matches an output y.

$$y = h(x). \quad (1)$$

In [16] Tejashri J. Joshi et al. try to classify extracted features using minimum Euclidean Distance between the feature vectors of test image and training image. The correlation between continuous and quantitative variables are measured by Euclidean distance and it is not appropriate for ordinal data, where the preferences are listed by rank instead of depending on the actual and that reduces the accuracy. To fix this problem they tried to use K-means Clustering as simple classifier. All features of the same class are combined to form a cluster. The minimum distance between the feature vector of the test image and cluster centers and is computed and the class of the test image is predicted. However this classifier is not robust against outliers and so it reduces accuracy. All those problems were solved by using the SVM classifier to solve multi-class problem. To overcome the problem of unclassified region caused by indirect methods of the multiclass SVM which are one-against-all, one-against-one, Directed Acyclic Graph "DAG" SVM, a decision tree based multiclass SVM was used. In [10] A. Saxena et al. decided to use hand tokens cited in the section above to train a feed-forward backpropagation neural network. The network has only one input layer, hidden layer and output layer to simplify and accelerate the calculations. In [8] Prasuhn used a database storing a set of hand gesture images and to overcome the problem of labeling and classifying correctly all degrees of freedom of human's hand image, images of database were synthesized by using an open source software named LibHand [17] which is a library for human hand articulation. Recognition method is based on finding the best-match image in the database. Each image stored in the database is pretreated and has a HOG feature. Here, the simplest brute force matching was used because the number of images in the database is low. HOG features of the test image is compared with each database image's HOG feature. The "L2" distance of two HOG features is used as "matching metric". Image in the database with the minimum "L2" distance is chosen as the best-matched image. In [14] Mahesh B. Mariappan et al. used Cascaded Haar Classifiers to efficiently track object (fist) by the mobile device. To collect training images, they shot a short video of a closed fist in different lighting conditions and from different angles for about two minutes at twenty five frames per second. Then a video editing program was used to extract images from video. After obtaining positive image samples and negative image samples in the training database a border around region of interest in each positive image was marked by using an object marking program. This generate an output file that contains all coordinates of positive images (the name of the image file, the width and the height of the object of interest, the coordinates (x,y) of the top-left corner of the object of interest) and the same thing was done for negative images. Finally the HaarTraining program was invoked after packing all positive images into a vector file using createsamples program which output a.vec file. Using this vector file and the file containing all data about negative images HaarTraining program generate a directory full of training data. Finally Convert_Cascade program was used to converts the training data into an XML file which represents the final output of the Haar training process and which can be used for recognition process.

3 Application Domains of Hand Gesture Recognition for Mobile Devices

This section gives a brief overview of some of the advanced application areas of vision based hand gesture recognition systems for mobile devices. Hand gesture recognition systems for mobile device are applied in different applications on several areas, it includes interpreting sign language, device control, numbers recognition, gaming, etc. An overview of some areas of hand gesture recognition application for mobile devices is given below.

3.1 Numbers Recognition

Counting numbers and digit using gestures in a mobile device can be used to place orders to access some apps or to control something. H. Lahiani et al. [5, 6] proposed a system that uses the numbering to count fingers and that could be used in different manner as controlling the smartphone or a connected device. After interpreting the hand pose sign the orders could be given to the device to perform a special task. A. Saxena et al. [10] and J.L. Raheja et al. [11] developed a system that recognize digit from one to five. In [16] Tejashri J. Joshi et al. developed also a system that recognize numbers from 1 to 5.

3.2 Sign Language Recognition

Because sign language is used for interpreting hand signs made by people with special needs it has received special attention. Many systems have been proposed to recognize hand poses using different types of sign languages. For example, Setiawardhana et al. [13] system recognized Indonesian Sign Language. In [12] S. Swamy et al. developed a system that recognize Indian Sign Language. The proposed system by [8] Prasuhn et al. is applied to recognize American Sign Language.

3.3 Gaming and Augmented Reality

Another recent application of hand gesture is control games with the 3D modeling. In [14] Mahesh B. Mariappan et al. developed a system named “PicoLife” which constitutes an augmented reality game in which 3D characters are controlled by hand gestures in an Android smartphones.

3.4 Device Control

In [7] V. Shirshir Reddy et al. developed a system based on finger detection to control tablets. It is a virtual touchscreen controlled through hand movement and it provides a suitable efficient and user friendly interface between human and tablet.

In [18] T. Marasovic et al. developed an accelerometer based (but non-vision based) gesture recognition system to control mobile devices. The system was designed to run in real-time. The application is designed for Android operating system which uses the data from a single triaxial accelerometer to recognize nine different hand gestures.

4 Summary and Prospects

The table below shows summary of some hand gesture recognition systems for mobile devices. In Table 1 a comparison between different systems is made and in which we give a summary of different extraction methods and classifiers used by system.

Table 1. Comparison between hand gesture recognition methods for mobile device

Researchers	Image processing	Extracted features	Classifier	Recognition rate
[5] H. Lahiani et al. (2015)	skin color segmentation	Finger vectors	SVM	93%
[8] L. Prasuhn et al. (2014)	binarization in HUV color space	HOG feature	Brute force matching	47%
[10] A. Saxena et al. (2014)	Sobel Edge Detector	Hand tokens	feed-forward back propagation neural network	77%
[13] Setiawardhana et al. (2015)	skin color segmentation	Haar features for detection And hand shape for recognition	Viola-Jones for detection and K-nearest neighbor (k-NN) for recognition	*100% if distance < 50 cm *25% if distance = 75 cm
[12] S. Swamy et al. (2014)	HSV model's threshold value	LBP features	AdaBoost (Viola-Jones)	62%
[16] Tejashri J. Joshi et al. (2015)	Thresholding, reorientation, cropping, normalization	Principal Component Analysis (PCA)	*Euclidian distance *K-mean cluster *SVM	*72.4% for Euclidian distance *45% for K-mean cluster * 97.6%. for SVM

Table 1 shows that vision based hand gesture recognition for mobile devices is a growing with promising results and especially with the rise of the use of smartphones and tablets. A possible future work is possible to improve existing works such as hybridization between artificial vision and the use of sensors and accelerometer which already exist in the majority of todays' mobile devices. This may make systems more robust.

5 Conclusion

In this work various methods are discussed for hand gesture recognition for mobile devices, these methods include Color based segmentation, Edge detection, etc. for image processing, and PCA, LBP feature, Haar feature, etc. for features extraction and Neural Network, SVM, Viola-Jones Algorithm, etc. for classification. Also application domains for those systems are presented. A literature review and comparison of recent recognition systems are given as well. Summary of some hand gesture recognition systems are listed and a possible future work is presented as well.

References

1. Rautaray, S.S., Agrawal, A.: Vision based hand gesture recognition for human computer interaction: a survey. Artif. Intell. Rev. **43**, 1–54 (2012)
2. Stergiopoulou, E., Papamarkos, N.: Hand gesture recognition using a neural network shape fitting technique. Elsevier Eng. Appl. Artif. Intell. **22**(8), 1141–1158 (2009)
3. Wu, Y., Liu, Q.: An adaptive self-organizing color segmentation algorithm with application to robust real-time human hand localization. In: Proceedings of Asian Conference on Computer Vision, Taiwan (2000)
4. Yadav, D.K., Sharma, L., Bharti, S.K.: Moving object detection in real-time visual surveillance using background subtraction technique. In: 14th International Conference on Hybrid Intelligent Systems (HIS), pp 79–84 (2014)
5. Lahiani, H., Elleuch, M., Kherallah, M.: Real time hand gesture recognition system for android devices. In: 15th International Conference on Intelligent Systems Design and Applications (ISDA), pp. 592–597 (2015)
6. Lahiani, H., Elleuch, M., Kherallah, M.: Real time static hand gesture recognition system for mobile devices. J. Inf. Assur. Secur. **11**, 67–76 (2016). ISSN 1554-1010
7. Reddy, V.S., Raghuveer, V., Krishna, J.V., Chandralohit, K.: Finger gesture based tablet interface. In: IEEE International Conference on Computational Intelligence and Computing Research (ICCIC), pp. 1–4 (2012)
8. Prasuhn, L., Oyamada, Y., Mochizuki, Y., Ishikawa, H.: A HOG-based hand gesture recognition system on a mobile device. In: IEEE International Conference on Image Processing (ICIP), pp. 3973–3977 (2014)
9. Szeliski, R.: Imageprocessing, 1st edn. Springer, New York (2010). chapter 3, pp. 112–113
10. Saxena, A., Jain, D.K.: Hand gesture recognition using an android device. In: Fourth International Conference on Communication Systems and Network Technologies, pp. 819–822 (2014)
11. Raheja, J.L., Singhal, A.: Android based portable hand sign recognition system. Cornell University Library (2015)
12. Swamy, S., Chethan, M.P., Karnataka, M.: Indian sign language interpreter with android implementation. Int. J. Comput. Appl. **97**(13), 36–41 (2014)
13. Setiawardhana, S., Hakkun, R.Y., Baharuddin, A.: Sign language learning based on android for deaf and speech impaired people. In: 2015 International Electronics Symposium (IES), pp. 114–117 (2015)
14. Mariappan, M.B., Guo, X., Prabhakaran, B.: PicoLife: a computer vision-based gesture recognition and 3D gaming system for android mobile devices. In: 2011 IEEE International Symposium on Multimedia (ISM), pp. 19–26 (2011)

15. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2005, vol. 1, pp. 886–893 (2005)
16. Joshi, T.J., Kumar, S., Tarapore, N.Z., Mohile, V.: Static hand gesture recognition using an android device. Int. J. Comput. Appl. **120**(21), 48–53 (2015). (0975–8887)
17. Saric, M.: Libhand: a library for hand articulation. Version 0.9 (2011)
18. Marasovic, T., Papic, V.: User-dependent gesture recognition on android handheld devices. In: 22nd International Conference on Software, Telecommunications and Computer Networks (SoftCOM) (2014)
19. Lahiani, H., Kherallah, M., Neji, M.: Hand pose estimation system based on Viola-Jones algorithm for android devices. In: 13th ACS/IEEE International Conference on Computer Systems and Applications, (AICCSA) (2016)

1D Signal Processing for Improvement of People Counting Estimation Results

Sumaiyya Farooq(✉), Shoab Ahmed Khan, and M. Usman Akram

Department of Computer Engineering, College of E&ME,
National University of Sciences and Technology, Islamabad, Pakistan
sumaiyya20@hotmail.com, kshoab@yahoo.com, usmakram@gmail.com

Abstract. The paper proposes a novel method to overcome the problem of occlusion while estimating the density of pilgrims in highly congested scenarios. Millions of Muslims gather at Al-Haram Mosque every year to perform Hajj and Umrah. Therefore, the need to develop a robust computer aided system for the estimation of density of pilgrims for averting stampedes can not be denied. The proposed method uses the number of motion vectors with non-zero magnitude for designing a linear regression model to estimate the density of pilgrims. There are several challenges associated with the development of such a system. Therefore the number of motion vectors do not always linearly correlate with the corresponding number of pilgrims. The property of inter-frame correlation is exploited to maintain linearity between the number of motion vectors and people count using smoothing filter from 1D signal processing techniques. Experimental results show that the Percentage Mean Absolute Relative Error (PMARE) of the proposed system is nearly 2.5%.

Keywords: Motion vectors · Linear regression · Smoothing · People counting

1 Introduction

The Holy Mosque in Makkah attracts one of the largest crowds in the world. Pilgrims gather there in millions every year to perform the rituals of Hajj and Umrah. A huge number of security staff is required in the mosque for the purpose of crowd management, but advances in machine learning, computer vision and image processing are providing alternative solutions to monitor and manage the crowd using the video stream from the surveillance cameras installed around the mosque.

Several techniques have been presented in literature to count the number of people in crowded scenes. These techniques are broadly classified into direct and indirect approaches according to Sami et al. in [1]. Direct approach segments out persons from the crowded scenes before counting them and is therefore complex to implement. Head detection method proposed by [2], head detection and

A. Abraham et al. (eds.), *Proceedings of the 16th International Conference on Hybrid Intelligent Systems (HIS 2016)*, Advances in Intelligent Systems and Computing 552,
DOI 10.1007/978-3-319-52941-7_32

tracking of people using geometry of the human body [3] and head and shoulder shape detection method presented by [4] have been used by several researchers to segment out persons. Variation in environmental conditions, for example, illumination variation, shadows of objects, non-rigid shape of human body and occlusion are some of the big challenges of the direct approach [1].

Indirect approach works by extracting features of crowded scenes to estimate person count. The features may include pixel level information for example area of foreground pixels [5], texture level information [6] or other interest points like corners or edges or histogram of direction. Because of the relative simplicity of indirect method, they are used quite often for the crowd estimation systems.

Hajer Fradi and Jean-Luc Dugelay proposed a people counting system in [7]. They used foreground pixel counts and corner density as frame-based features. Gaussian process regression was used to make estimation of people count on the basis of these features. To address the issue of occlusion in crowded scenes, motion model was incorporated in the GMM background estimation to accurately segment the foreground region. Weights were assigned to the foreground pixels to normalize the distortions due to perspective variation.

A.B. Chan et al. proposed a system to count people in inhomogeneous crowded scenarios using indirect approach [8]. The crowd was divided into various components based on the tracking of features extracted from the homogeneous motion of the components. Two models, Gaussian process and Poisson regression, were used to estimate the count of people from the extracted features.

Zheng Ma and Antoni B. Chan [9] came up with an integer programming method in which they used Line of Interest (LOI) approach to count the number of people. The video sequence was temporally divided into chunks and low level features were extracted. These features were mapped to the number of people in the ROI using regression model. Using this people count information, number of people were estimated crossing the LOI. Junliang Xing et al. in [10] performed people detection method taking multiple frames of the video in temporal sequence. The purpose of using multiple frames was to address the problem of miss and false detections and occlusions. In addition, further information about the direction of the motion of crowd was obtained.

The authors highlighted an optical flow based approach for the estimation of crowd size in [11]. After calculating optical flow on the image slices, blobs were detected and their features such as velocity, position and size of the blob were extracted. The number of people were counted using the regression technique. Rao et al. in their work [12] proposed a block-based dense optical flow method with spatial and temporal filtering to calculate the velocities. These velocities were used to get the location of objects in crowded scenarios. Moreover, hierarchical clustering was carried out to cluster the objects based on Euclidean distance metric.

To work with Hajj data involves many challenges because of high congestion level that ultimately leads to the occlusion of pilgrims. Moreover, illumination variation, shadows of moving pilgrims and low contrast between the clothing of pilgrims and the white marble floor [13] further increases the difficulty to the

pilgrims counting system. In our work, we have proposed an indirect method for the estimation of number of pilgrims using motion estimation approach. To make the counting results more accurate, frame-to-frame correlation has been developed in the features that are linearly regressed to count the number of pilgrims in the crowded scenarios in Hajj and Umrah events.

The paper is organized in the following sequence: Sect. 2 describes the proposed methodology. Section 3 presents the results of the proposed framework. In Sect. 4, discussion of results is presented and Sect. 5 concludes the paper.

2 Proposed Methodology

Proposed system is targeted towards improving the results of density estimation of pilgrims by exploiting frame-to-frame correlation. Figure 1 shows the top level diagram of the proposed method. Because of unavailability of standard dataset, we have used a nonstandardized video sequence captured at one of the entrance gates of Al-Haram mosque, Makkah. The camera is assumed to be stationary and mounted at a raised position. In this way the effect of occlusion is implicitly suppressed to some extent. The dataset is manually annotated to set the ground truth of pilgrims count in the frames.

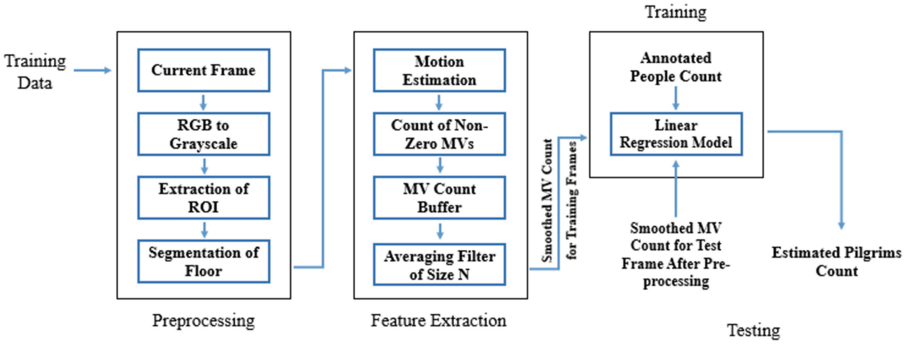


Fig. 1. Flow of proposed framework for density estimation of pilgrims.

2.1 Preprocessing Steps

The preprocessing block in Fig. 1 consists of many steps that are applied on the input video frames. These steps are discussed one by one below:

1. Incoming frame is converted from RGB domain to grayscale because the three channels contain the same information content.
2. Region of interest (ROI) is defined by simply cropping the input frame because the entire frame carries some redundant information that does not need to be processed. This redundancy includes the light poles and building of the mosque. ROI includes only that part of the frame which contains moving pilgrims.

3. Next step is the segmentation of pilgrims from the floor which is always static. Moving pilgrims cast their shadows on the floor which also move with them. To differentiate between the true motion of pilgrims and the false motion of the shadows, an adaptive binary segmentation method is developed. This segmentation method includes the following steps:
   (a) Input frame is divided into equal sized blocks of 50×50 pixels.
   (b) The adaptive threshold level for segmentation is taken as 80% of maximum intensity value of the block and segmentation is performed using the following equation. The binary mask is then complemented.

$$BinaryMask = \begin{cases} 0; p(x,y)|_b < 0.8 * Imax|_b \\ 1; p(x,y)|_b \geq 0.8 * Imax|_b \end{cases} \quad (1)$$

   where;
   $p(x,y)|_b = PixelIntensityAtPosition(x,y)OfBlock'b'$
   $Imax|_b = MaximumIntensityValueOfBlock'b'$

2.2 Motion Estimation

A hybrid method consisting of block matching followed by optical flow has been proposed by S.H. Chan et al. in [14] and is used to calculate motion vectors of the frames. Integer part of motion vectors is computed using block matching technique. It uses Sum of Absolute Differences (SAD) as the matching criterion between consecutive frames. Block size 'N' is kept 8×8 and search window 'M' has a size 16×16. After calculating integer part, Taylor approximation is used to compute the decimal part of the vectors which indicates sub-pixel motion.

The binary mask calculated in Sect. 2.1 has been used to discard the false motion. This false motion results from the moving shadows on the white marble floor. Making white pixels i.e. 1s of the binary mask as a necessary condition, motion vectors have been computed only for those pixels which experience true motion.

A motion vector V with V_x as its x-coordinate and V_y as its y-coordinate has a magnitude given as:

$$V_{mag} = \sqrt{V_x^2 + V_y^2} \quad (2)$$

2.3 Linear Regression Model for Counting of Pilgrims

Direct methods can not be used for counting of pilgrims because of the occlusion of pilgrims. Majority of pilgrims wearing white Ahraam in general and women wearing black Abaya in particular appear as blobs while moving. Therefore, it is quite impossible to detect them as individuals on the basis of head detection or head and shoulder shape detection algorithms. Therefore, we have adopted the indirected method to estimate the number of pilgrims in our ROI. Number of motion vectors that have non-zero magnitude is used as the input feature to the linear regression model to make a quantitative estimation of density of pilgrims. The reason for using linear regression model is the obvious linear relationship

between the number of pilgrims in the scene and the motion vectors associated with them. In ideal case when there is no occlusion, number of motion vectors increase with the count of pilgrims and the other way round. The linear regression equation used for the estimation is given as:

$$\hat{C} = a * MV + b \tag{3}$$

where;
$\hat{C}$ = estimated pilgrim count
MV = number of motion vectors
a = rate of increase of decrease of $\hat{C}$ for unit increase in MV
b = y-intercept

2.4 Frame-to-Frame Correlation Using Smoothing Filter

Not so exact linear relationship between the number of motion vectors having non-zero magnitude and the respective number of pilgrims has been observed for such a congested scenario. The prime reason for this non-linearity is the high levels of occlusion. While moving, pilgrims overlap with each other across the frames and then move away due to which number of motion vectors do not increase or decrease in accordance with count of motion vectors. Red circles in Fig. 2 illustrate the scenario. Moving from left to right, pilgrims move towards each other leading to a decreased number of motion vectors and then depart bringing the number to the original value. The count of motion vectors does change but the total number of pilgrims in the sub-image remain approximately constant; thus invalidating the linear relationship.

To force the relationship between the two quantities to be linear, the property of frame-to-frame correlation has been exploited. Two consecutive frames can not have a significant variation with respect to people count. Therefore, to remove the noise from the motion vector count values, smoothing filter from 1D signal processing techniques has been applied to the values treating them as a 1D signal. Smoothing filter used in our work is a N-tap Moving Average Filter (MAF) (Fig. 3). The first location of the buffer holds the motion vector count for the current frame. Remaining N-1 locations contain count values for N-1 previous frames. The average value of all these counts replaces the motion vector count value for the current frame. The smoothed values of the counts for the training frames are used as the input of the linear regression model. For a test frame, the motion vector count value is again passed through the smoothing filter using N-1

Fig. 2. Pilgrims overlapping and separating in time.

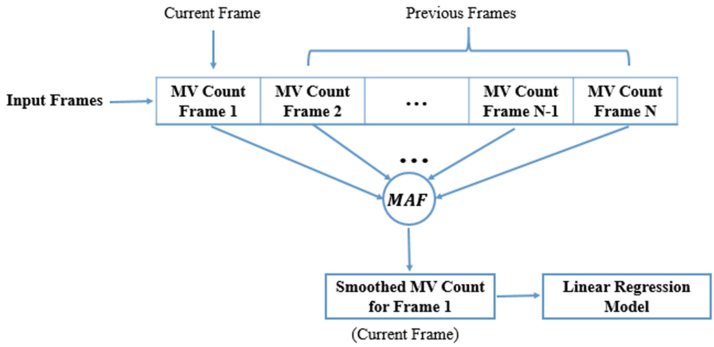


Fig. 3. MV count buffer for frame-to-frame correlation.

previous test frames and pilgrims count is estimated using the smoothed value of motion vector count. Value of N has been kept 3 on empirical basis.

In this way, the irregularities in the count of motion vectors between successive frames have been smoothed. Occlusion of pilgrims and impediment in accurate segmentation of pilgrims from their shadows due to low contrast between white floor and pilgrims wearing white Ahraam are the prime reasons of these spikes and dips.

3 Experimental Results

The performance of estimation system of pilgrims is evaluated using error measures, for example, Mean Absolute Error (MAE) and Percentage Mean Absolute Relative Error (PMARE). Histogram of Percentage Absolute Relative Error (PARE) is also plotted to observe the error distribution.

3.1 Preprocessing Steps

Figure 4 shows the original frame, gray scale frame, the extracted ROI and the binary mask calculated to segregate the pilgrims and their shadows. The red boundary in Fig. 4(D) discriminates the moving pilgrims from their moving shadows.

3.2 Motion Estimation

Motion vectors have been calculated using binary mask as a prerequisite to nullify the false motion vectors. In Fig. 5(A), motion vectors are visible on some parts of the floor. These have been cancelled out with the help of the binary mask condition. The cleaned up motion vector grid superimposed on the frame is shown in Fig. 5(B).

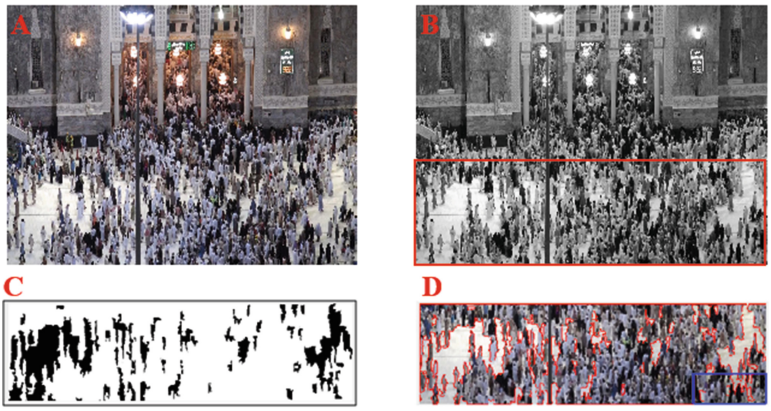


Fig. 4. Preprocessing steps (A) original frame (B) gray scale converted frame (C) binary mask of extracted ROI (D) segmentation of pilgrims from their shadows.

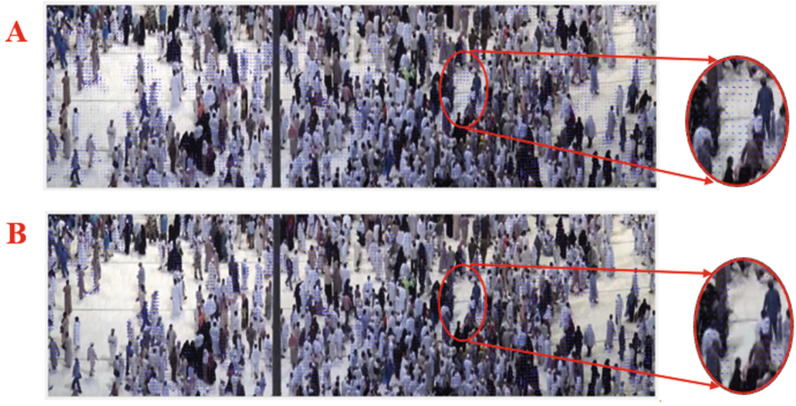


Fig. 5. (A) Motion vector grid without using binary mask (B) motion vector grid with using binary mask.

3.3 Linear Regression Model Using Frame-to-Frame Correlation for Counting of Pilgrims

Linear regression model is a category of supervised learning in which we need separate datasets for training and testing purposes. Dataset has been divided into two parts. First part has been used for training the linear model and the second part has been tested to evaluate the performance of the model.

To make the relationship between the number of non-zero motion vectors and the corresponding number of pilgrims linear, moving average filter has been used to remove the anomalies from the motion vector count in successive frames. After applying an 3-tap moving average filter to a very small subset of testing frames,

	Frame 44	Frame 45	Frame 46	Frame 47	Frame 48	Frame 49	Frame 50
Original MV Count →	7710	7664	7719	7670	7740	7796	7789
Smoothed MV Count →			7697	7684	7709	7735	7775

Fig. 6. Number of non-zero motion vectors versus pilgrims count before and after smoothing.

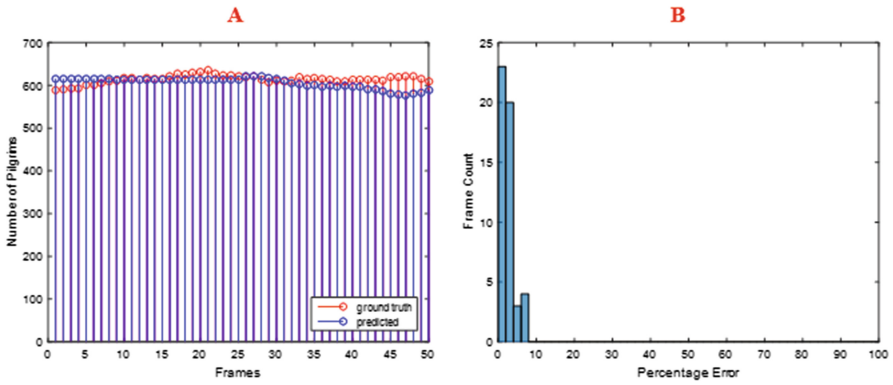


Fig. 7. (A) Estimated number of pilgrims (B) Histogram of Absolute Relative Error (ARE) of estimation.

the spikes and dips in the motion vector count values have been significantly suppressed as shown in Fig. 6.

Exploiting the temporal correlation in frames and using the linear regression model, number of pilgrims have been estimated for the testing set of frames. In the graph shown in Fig. 7(A), the estimation results for the testing frames have been presented. Red circles show the ground truth values whereas blue circles show the predicted count values of pilgrims. Figure 7(B) shows that maximum error is concentrated towards the left side i.e. less than 10%.

The number of pilgrims remain approximately 600 in our region of interest. With these many pilgrims, the performance error measures such as Mean Absolute Error (MAR) and Percentage Mean Absolute Relative Error (PMARE) are obtained as approximately 15 and 2.5% respectively.

4 Discussion

We have proposed a framework for the counting of pilgrims using motion estimation approach. 1D signal processing technique has been incorporated in order to improve the estimation results of pilgrims counting system. There are two main reasons for the counting error. The prime reason is the heavy occlusion of pilgrims. When pilgrims move, they overlap each other hiding the people

behind them. In some cases, the head from the entire body is barely visible of the occluded pilgrims. Therefore, there is a significant reduction in the motion vector count as can be seen in Fig. 6. To overcome this problem, smoothing filter has been used to develop correlation in successive frames.

Second reason is the variation in illumination conditions in the region under consideration. Due to change in lighting, the binary mask extracted for the segmentation of static floor and moving persons is not absolutely correct. This minute error leads to the anomalies in the estimation results.

To our best knowledge, not much work has been done yet for the development of a crowd estimation system for the specific events of Hajj and Umrah where millions of pilgrims gather every year. Moreover, no standard dataset of Hajj is available which can be used to build a crowd estimation system. Therefore, comparison in true sense can not be made between the existing techniques and our proposed methodology to count the number of pilgrims.

5 Conclusion

We have presented a novel idea to estimate the count of pilgrims in dense crowded areas and occluded situations. After carrying out preprocessing steps, motion in the scene is estimated using a combination of block matching and optical flow techniques. People counting based on detection of individuals is not possible because of highly congested situation. Therefore, number of motion vectors that have non-zero magnitude have been used as the potential indicator of pilgrims count. Based on the non-zero motion vector count of a frame, a linear regression model has been used to estimate the number of pilgrims. Because of occluding environment resulting from dense crowd, frame-to-frame temporal correlation has been exploited in order to improve the counting estimation results. In future, proposed methodology can be further extended to develop much complex decision support systems for the prevention of any fatal incidents.

References

1. Saleh, S.A.M., Suandi, S.A., Ibrahim, H.: Recent survey on crowd density estimation and counting for visual surveillance. Eng. Appl. Artif. Intell. **41**, 103–114 (2015)
2. Zhou, T., Yang, J., Loza, A., Bhaskar, H., Al-Mualla, M.: Crowd modeling framework using fast head detection and shape-aware matching. J. Electron. Imaging **24**(2), 023019 (2015)
3. Yaowu, H., Zhou, P., Zhou, H.: A new fast and robust method based on head detection for people-flow counting system. Int. J. Inf. Eng. **1**(1), 33–43 (2011)
4. Li, M., Zhang, Z., Huang, K., Tan, T.: Estimating the number of people in crowded scenes by mid based foreground segmentation and head-shoulder detection. In: 2008 19th International Conference on Pattern Recognition, ICPR 2008, pp. 1–4. IEEE (2008)
5. Hou, Y.-L., Pang, G.K.H.: People counting and human detection in a challenging situation. IEEE Trans. Syst. Man Cybern. Part A Syst. Hum. **41**(1), 24–33 (2011)

6. Marana, A.N., Cavenaghi, M.A., Ulson, R.S., Drumond, F.L.: Real-time crowd density estimation using images. In: Bebis, G., Boyle, R., Koracin, D., Parvin, B. (eds.) ISVC 2005. LNCS, vol. 3804, pp. 355–362. Springer, Heidelberg (2005). doi:10.1007/11595755_43
7. Fradi, H., Dugelay, J.-L.: Low level crowd analysis using frame-wise normalized feature for people counting. In: 2012 IEEE International Workshop on Information Forensics and Security (WIFS), pp. 246–251. IEEE (2012)
8. Chan, A.B., Vasconcelos, N.: Counting people with low-level features and Bayesian regression. IEEE Trans. Image Process. **21**(4), 2160–2177 (2012)
9. Ma, Z., Chan, A.B.: Crossing the line: crowd counting by integer programming with local features. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2539–2546 (2013)
10. Xing, J., Ai, H., Liu, L., Lao, S.: Robust crowd counting using detection flow. In: 2011 18th IEEE International Conference on Image Processing, pp. 2061–2064. IEEE (2011)
11. Benabbas, Y., Ihaddadene, N., Yahiaoui, T., Urruty, T., Djeraba, C.: Spatio-temporal optical flow analysis for people counting. In: 2010 Seventh IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), pp. 212–217. IEEE (2010)
12. Rao, A.S., Gubbi, J., Marusic, S., Stanley, P., Palaniswami, M.: Crowd density estimation based on optical flow and hierarchical clustering. In: 2013 International Conference on Advances in Computing, Communications and Informatics (ICACCI), pp. 494–499. IEEE (2013)
13. Al-Khaffaf, H.S.M., Haron, F., Sarmady, S., Talib, A.Z., Abu-Sulyman, I.M.: Crowd parameter extraction from video at the main gates of Masjid al-Haram. In: Sambath, S., Zhu, E. (eds.) Frontiers in Computer Education. Advances in Intelligent and Soft Computing, vol. 133, pp. 727–736. Springer, Heidelberg (2012). doi:10.1007/978-3-642-27552-4_96
14. Chan, S.H., Nguyen, T.Q., et al.: Subpixel motion estimation without interpolation. In: 2010 IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 722–725. IEEE (2010)

Intelligent Control Strategy of a Three-Phase PWM Rectifier Based on Artificial Neural Networks Approach and Fuzzy Logic Controller

Mustapha Jamma[1(✉)], Mohamed Barara[1,2], Abderrahim Bennassar[1], and Mohammed Akherraz[1]

[1] Mohammadia School's of Engineers, Mohammed V University, Rabat, Morocco
jamaa120@hotmail.com
[2] Université de Lyon, F-69622, Université Claude Bernard Lyon 1, Villeurbanne; CNRS; UMR 5005, Laboratoire Ampère, Lyon, France
http://www.emi.ac.ma

Abstract. This paper proposes a new direct power control (DPC) of three-phase PWM rectifier based on intelligent techniques, in order to improve the dynamic performances of the conventional direct power control, where the conventional proportional integral (PI) controller is replaced by fuzzy logic controller (FLC) to adjust the dc bus voltage and the classical switching table is also replaced by a selector based on artificial neural networks (ANN) to generate the switching sequences of the PWM rectifier. This new control approach allows to maintain the dc bus voltage, the instantaneous active and reactive power to their reference values and also to minimize their ripples. The dynamic performances of this control technique were verified using Matlab/Simulink software under different conditions of simulation. The obtained results present better performance in terms of precision, robustness and reduction of harmonic disturbances.

Keywords: Fuzzy logic controller · Three-phase PWM rectifier · Artificial neural networks · Direct power control

1 Introduction

The static converters of power electronics, based on diode/thyristor bridge circuit, have been widely used in industrial and domestic areas, due to its reliability, simplicity and high performance compared to conventional electromechanical converters [1]. However, these main disadvantages are the low power factor and the harmonic pollution of electrical distribution grid. These harmonics allow the deterioration of the quality of current and voltage wave of grid. They can even train a bad exploitation of energy as well as the dysfunction at the level of the electrical appliances. These constraints have directed the researchers to propose

A. Abraham et al. (eds.), *Proceedings of the 16th International Conference on Hybrid Intelligent Systems (HIS 2016)*, Advances in Intelligent Systems and Computing 552,
DOI 10.1007/978-3-319-52941-7_33

more effective solutions using new fully controllable power components able to avoid the appearance of harmonic pollution or to limit their propagation. Among these solutions, we find the three phase PWM rectifiers. This type of converter has several advantages such as, the control of the absorbed reactive power and the consumption of a current near to a sinusoid by reducing its harmonic content and the size of filters and so on [2,3].

Various control structures of the PWM rectifiers have been developed thanks to developments of digital technologies and the emergence of new power components [4]. They can be classified, according to their principles into four categories: voltage-based direct power control (V-DPC), virtual-flux-based direct power control (VF-DPC), voltage oriented control (VOC) and virtual-flux oriented control (VFOC) [5]. Indeed the overall goal of these strategies is to get an exact control of the currents and powers as well as a low total harmonic distortion (THD). In addition, the VOC is based on the control loops of the currents in the d-q synchronous frame [4]; it guarantees a high dynamic and static behavior as well as a fast transient response [5]. Besides, the DPC is based on control loops of instantaneous powers and on position of the grid voltage [6]. It is equivalent to the direct torque control (DTC) for the electrical machines [7]. However, its advantages are a simple algorithm, no current regulation loops and good dynamic performance.

The application of ANN have been reported in literature, for example in optimization of the wind power capture [8], also the fuzzy logic is used for regulating the dc voltage as presented by [9] and also it has been used to control the speed of induction motor [10]. The combination of these two intelligent regulators can improve the effectiveness of the control system.

In this work, we present a new direct power control for a three-phase PWM rectifier in order to eliminate the harmonic currents, to control the instantaneous powers and to reduce their ripples. This new direct power control consists, on one part, to replace the conventional switching table by a selector based on artificial neural networks in order to reduce considerably the ripples of instantaneous active and reactive power, and on the other part, in controlling the dc bus voltage using a fuzzy logic controller.

2 The Mathematical Model of Three-Phase PWM Rectifier

The schematic diagram of a three phase PWM rectifier is illustrated in Fig. 1. It is composed of six IGBTs and six diodes. (e_a, e_b, e_c) and (i_a, i_b, i_c) represent respectively the three line voltages and three line currents; R and L are respectively the line resistance and the line inductance; C is the filter capacitance of the dc bus voltage; R_L is the resistive load; i_r is rectified current and i_L represents the current of the load.

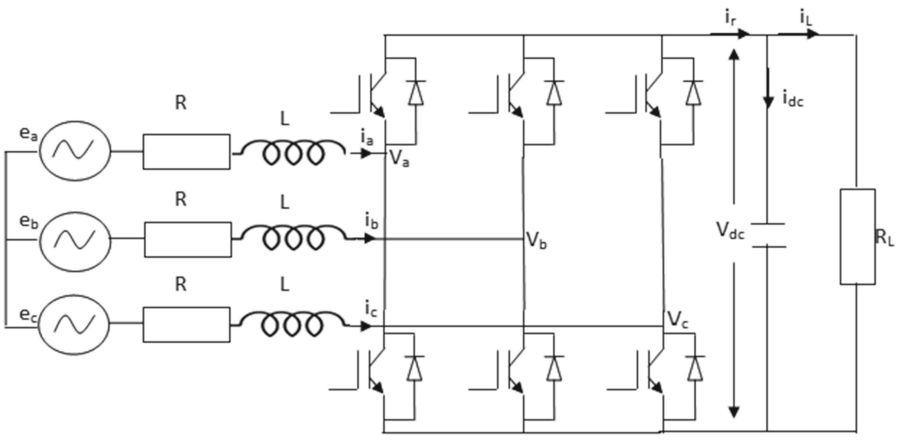


Fig. 1. The schematic diagram of a PWM rectifier.

The equations describing the model of the three-phase PWM AC-DC converter in the stationary $\alpha\beta$-coordinates are given by:

$$\begin{cases} \frac{di_\alpha}{dt} = \frac{1}{L}(e_\alpha - v_\alpha - R.i_\alpha) \\ \frac{di_\beta}{dt} = \frac{1}{L}(e_\beta - v_\beta - R.i_\beta) \\ C\frac{dv_{dc}}{dt} = S_\alpha.i_\alpha + S_\beta.i_\beta - i_L \end{cases} \quad (1)$$

With (i_α, i_β) and (S_α, S_β) represent respectively the line currents and the rectifier switching states in stationary $\alpha\beta$-coordinates system.

3 The Principle of the Proposed DPC

3.1 System Configuration

The DPC is the control structure that directly uses the instantaneous active and reactive power as control variables. The switching sequences of the switches of the bridge rectifier have been determined using a neural selector. The inputs of this selector are the sector where is the position of the voltage vector of the grid and the powers errors provided by hysteresis controllers [11]. The regulation loop of the dc bus voltage is based on fuzzy controller in order to maintain this voltage at its reference value. It is similar to the DTC for induction machines in which the torque and the stator flux are the quantities controlled. The simplified representation of the new DPC for a three-phase PWM rectifier is illustrated in Fig. 2.

To achieve a unity power factor (UPF), the reference of reactive power is set to zero. So, the DPC strategy uses the angular position of the voltage vector of the grid $e_{\alpha\beta}$ to determine the sector of work, for this, the vectors plane $\alpha\beta$

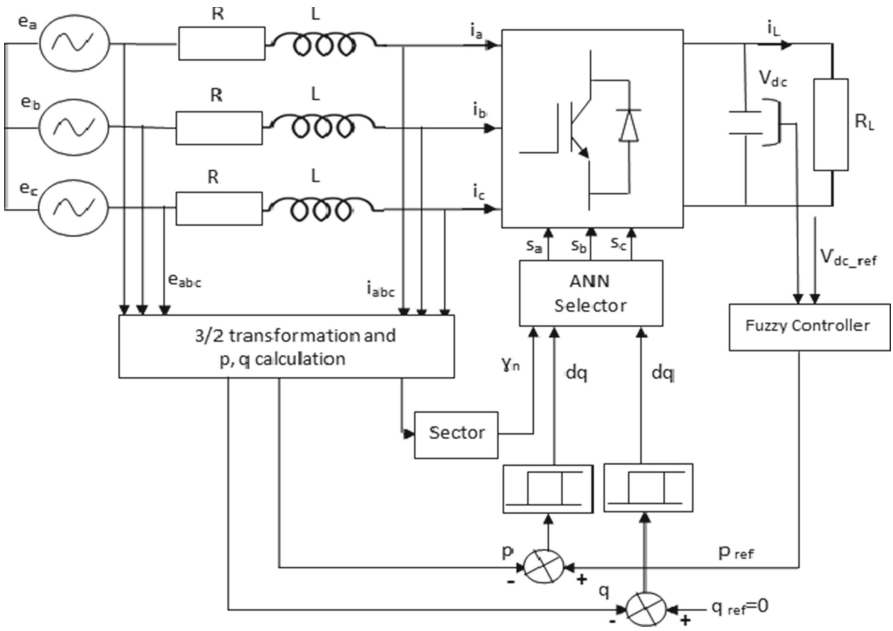


Fig. 2. Simplified representation of DPC for a three-phase PWM rectifier.

is divided into twelve sectors as shown in Fig. 3. These sectors are expressed numerically by this expression:

$$(n-2)\frac{\pi}{6} \leq \gamma_n \leq (n-1)\frac{\pi}{6} \;\; ; \;\; n = 1, 2, ..., 12 \tag{2}$$

where n is the sector number.

To establish the switching modes of the PWM rectifier, we use two hysteresis comparators at two levels which enable to establish two logical outputs d_p and d_q from the errors of the measured values and the reference values of the instantaneous active and reactive powers, by respecting the two following expressions:

$$dp = \begin{cases} 1, p_{ref} - p \geq H_p \\ 0, p_{ref} - p \leq -H_p \end{cases} \tag{3}$$

$$dq = \begin{cases} 1, q_{ref} - q \geq H_q \\ 0, q_{ref} - q \leq -H_q \end{cases} \tag{4}$$

H_p and H_q represent the deviations of hysteresis regulators. The switching table of DPC is given by the Table 1. The digitized values d_p, d_q and the sector of work γ_n are the inputs of this table, where the switching states of the three-phase PWM rectifier for all sectors are stored.

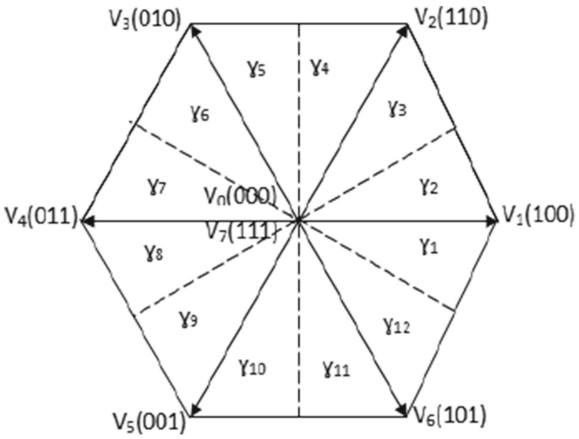


Fig. 3. Sectors and voltage vectors of three-phase PWM rectifier.

Table 1. Switching table of the DPC.

d_p	d_q	γ_1	γ_2	γ_3	γ_4	γ_5	γ_6	γ_7	γ_8	γ_9	γ_{10}	γ_{11}	γ_{12}
1	0	V_6	V_7	V_1	V_0	V_2	V_7	V_3	V_0	V_4	V_7	V_5	V_0
1	1	V_7	V_7	V_0	V_0	V_7	V_7	V_0	V_0	V_7	V_7	V_0	V_0
0	0	V_6	V_1	V_1	V_2	V_2	V_3	V_3	V_4	V_4	V_5	V_5	V_6
0	1	V_1	V_2	V_2	V_3	V_3	V_4	V_4	V_5	V_5	V_6	V_6	V_1

3.2 Determination of Instantaneous Powers

For a three phase system, the instantaneous active and reactive power may be determined by several techniques [11], such as measurement of line currents and line voltages. Using the complex notation, these powers are expressed respectively as follows:

$$p = R_e(e.i^*) \tag{5}$$

$$q = I_m(e.i^*) \tag{6}$$

where * denotes the conjugate of the line current vector.
In stationary $\alpha\beta$ coordinates, the instantaneous active and reactive power is given by the following expressions:

$$\begin{cases} p = e_\alpha.i_\alpha + e_\beta.i_\beta \\ q = e_\beta.i_\alpha - e_\alpha.i_\beta \end{cases} \tag{7}$$

4 The Proposed Neural Networks Selector

The development of neural networks is relatively recent. The origin of these latter comes from the modeling test of biological neuron [12]. They form a set of

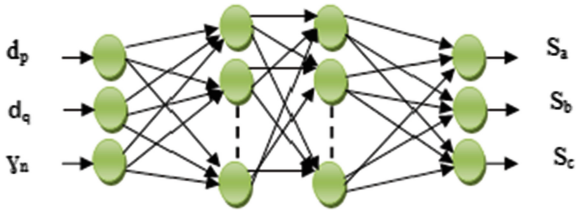


Fig. 4. Architecture of ANN selector.

Table 2. Parameters of the neural networks.

Number of neurons in the input layer	3 neurons
Number of neurons in the hidden layer	20 neurons
Number of neurons in the output layer	3 neurons
Number of epochs	1500
Mean square error	10^{-6}
Training algorithm of network	Backpropagation algorithm
Type of activation functions	Tansig and Purelin

nonlinear functions allowing building by learning, a vast family of models and non-linear correctors. The information in the neural networks spreads from one layer to another. We can distinguish three types of layers: an input layer, hidden layers and the output layer [13].

The neural networks are used in many areas, include the classification, pattern recognition, the static or dynamic modeling of process and the control of industrial processes [14].

In this work, we proposed to change the conventional selector of the switching sequences of the PWM rectifier by a neural networks selector, in order to reduce the ripples of active and reactive power. The inputs of the neural selector are the angular position of the voltage vector and the errors of instantaneous powers provided by the hysteresis comparators. So, these outputs are the switching states of the rectifier. Figure 4 shows the architecture of the neural networks used.

Table 2 shows the parameters of the neural networks used. Consequently, this neural selector is generated by Matlab/Simulink software.

5 Fuzzy Controller of the dc Bus Voltage

Since several years, the fuzzy logic controller has been of great topical. Indeed, this technique allows obtaining a very effective control law without having to do extensive modelling [15]. Figure 5 shows the block diagram of the fuzzy logic controller used to control the dc bus voltage of a three-phase PWM rectifier.

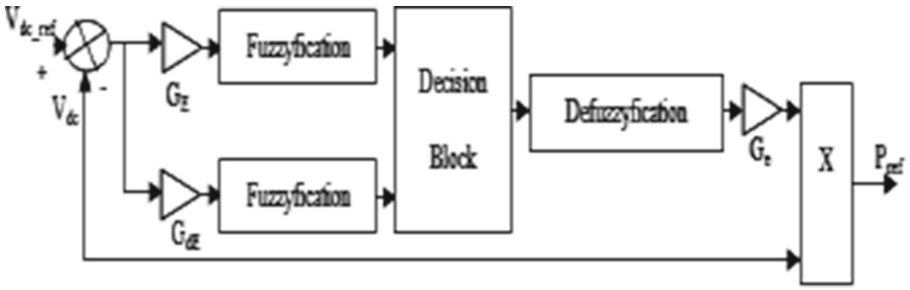


Fig. 5. Schematic diagram of fuzzy controller.

This fuzzy control has been designed using the method of fuzzy implication of Mamdani, founded on the min-max decision. It comprises three main blocks: Fuzzification, decision block and defuzzification [15].

In the system of the Fig. 5, the fuzzy controller inputs are the error E and its derivative dE. They are defined as follows:

$$\begin{cases} E(k) = v_{dc\_ref}(k) - v_{dc}(k) \\ dE(k) = E(k) - E(k-1) \end{cases} \tag{8}$$

The output of the fuzzy controller is multiplied by the measured value of the dc bus voltage in order to obtain the reference of the active power. The membership functions of the input and output variables are illustrated in Fig. 6. The seven fuzzy sets chosen to accomplish the fuzzification and defuzzification are: NB (Negative Big), NM (Negative Medium), NS (Negative Small), Z (Zero), PS (Positive Small), PM (Positive Medium) and PB (Positive Big).

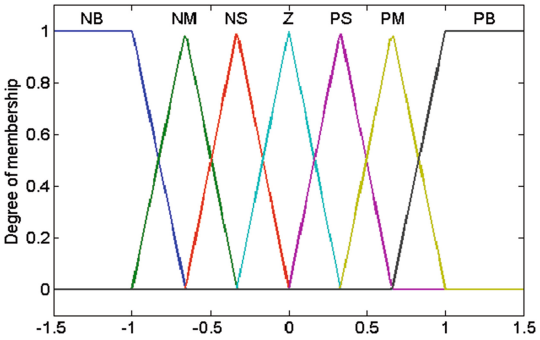


Fig. 6. Membership functions of inputs/outputs variables.

The rules that we have established are presented in Table 3:

Table 3. Inference rules.

dE\E	NB	NM	NS	Z	PS	PM	PB
PB	Z	PS	PM	PB	PB	PB	PB
PM	NS	Z	PS	PM	PB	PB	PB
PS	NM	NS	Z	PS	PM	PB	PB
Z	NB	NM	NS	Z	PS	PM	PB
NS	NB	NB	NM	NS	Z	PS	PM
NM	NB	NB	NB	NM	NS	Z	PS
NB	NB	NB	NB	NB	NM	NS	Z

6 Simulation and Analysis

In order to test the performance of the new direct power control strategy applied to a three-phase PWM rectifier, numerical simulations have been carried in the same conditions using Matlab/Simulink software. These results were performed for a DPC based on fuzzy control of dc bus voltage and a selector based on artificial neural networks. The parameters of the simulated system are indicated in Table 4.

The different results obtained by simulation under a power supply purely sinusoidal, balanced and with a unit power factor are illustrated in Figs. 7, 8 and 9. Indeed, Fig. 7 shows the simulation results of the new DPC for a constant reference voltage $V_{dc\_ref} = 600\ V$ and for a constant resistive load $R_L = 100\ \Omega$.

The line current i_a and the line voltage e_a are in phase (FPU) (Fig. 7(a)), the instantaneous active and reactive power are well controlled to their references (Fig. 7(d)). We also note that the dc bus voltage has a good tracking of their

Table 4. Parameters of the simulated system.

Parameters of circuit	Value
Sampling frequency	10 kHz
Line resistance R	0.25 Ω
Line inductance L	0.01 H
Load resistance R_L	100 Ω
dc-bus capacitor C	2400 μF
Grid phase voltage E	220 V
Source voltage frequency f	50 Hz
dc-bus voltage V_{dc}	600 V

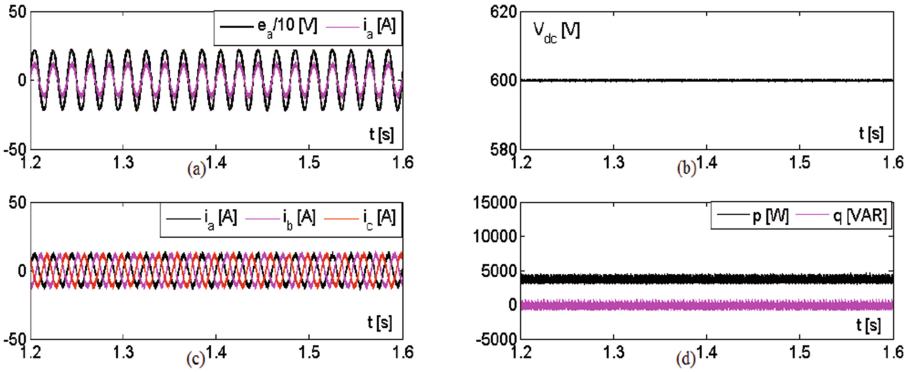


Fig. 7. Simulation results of PWM rectifier with new DPC. $V_{dc\_ref} = 600\ V$; $R_L = 100\ \Omega$; $q_{ref} = 0\ VAR$ (a) phase grid voltage and grid currents at UPF; (b) dc-link voltage waveform; (c) three-phase line currents; (d) instantaneous active and reactive power.

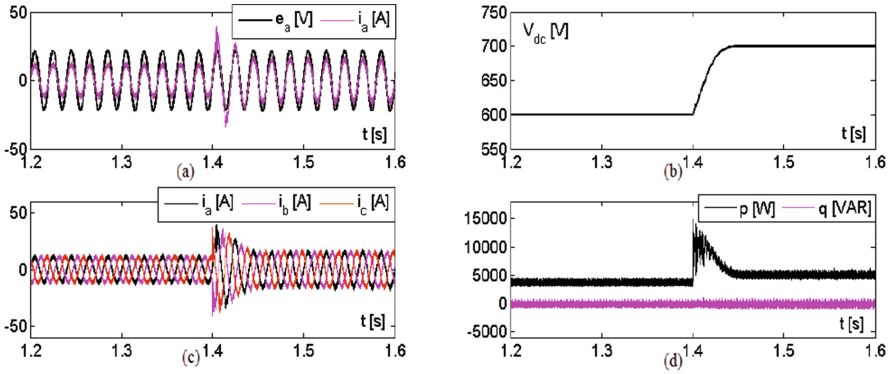


Fig. 8. Simulation results of PWM rectifier with new DPC at step change of the reference voltage ($V_{dc\_ref} = 600\ V$ to $V_{dc\_ref} = 700\ V$); $R_L = 100\ \Omega$, $q_{ref} = 0\ VAR$. (a) phase grid voltage and grid currents at UPF; (b) dc-link voltage waveform; (c) three-phase line currents; (d) instantaneous active and reactive power.

reference (Fig. 7(b)) and the three line currents are nearly sinusoidal with a reduced total harmonic distortion (THD=2.16 %) (Fig. 3(c)).

Figure 8 represents the waveforms obtained in the case of a variation of the reference voltage of $V_{dc\_ref} = 600\ V$ to $V_{dc\_ref} = 700\ V$ at $t = 1.4\ s$. In the case where the load is constant $R_L = 100\ \Omega$, this causes an increase in the line current (Fig. 8(c)) and active power without affecting the reactive power (Fig. 8(d)). It is to be noted that the DPC based on the intelligent techniques provides a good response, a good robustness, a reduction of the ripples of instantaneous powers and excellent control of the dc bus voltage (Fig. 8(b)).

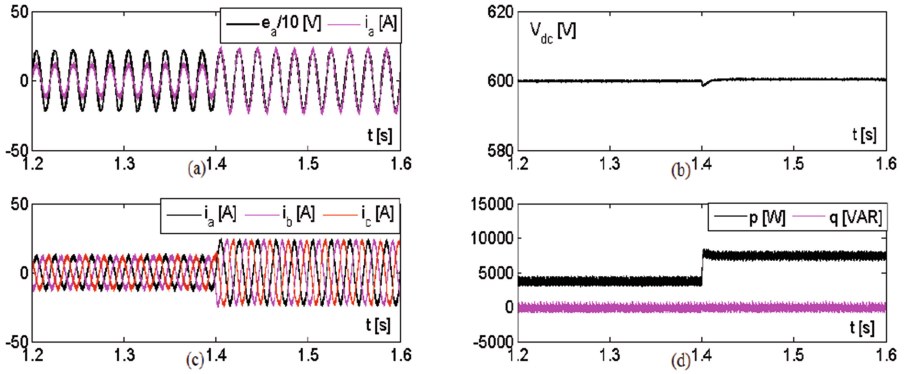


Fig. 9. Simulation results of PWM rectifier with new DPC at step change of the load ($R_L = 100\ \Omega$ to $R_L = 50\ \Omega$); $V_{dc\_ref} = 600\ V$; $q_{ref} = 0\ VAR$. (a) phase grid voltage and grid currents at UPF; (b) dc-link voltage waveform; (c) three-phase line currents; (d) instantaneous active and reactive power.

The waveforms obtained in the case of a decrease of the load of $R_L = 100\ \Omega$ to $R_L = 50\ \Omega$ at $t = 1.4\ s$, in the case where the reference voltage is constant $V_{dc\_ref} = 600\ V$, are shown in Fig. 9. Thanks to this load variation, the line current undergoes an increase (Fig. 9(c)), this causes an increase in active power (Fig. 9(d)), while the dc bus voltage is maintained at its reference after a very short duration of establishment (Fig. 9(b)).

During the simulation tests performed, we noticed that the DPC of a three-phase PWM rectifier based on a neural selector with regulation of the dc bus voltage by a fuzzy logic controller offered a good performance and a good rejection of harmonic disturbances.

7 Conclusion

A fuzzy logic dc bus voltage controller of a three-phase PWM rectifier with artificial neural networks selector for direct power control is presented in this work. This technique of controller is simulated by using Matlab/Simulink environment. The main objectives of the proposed control are to reduce the ripples of the instantaneous active and reactive power, maintain dc-bus voltage to the required level as well as it guarantees sinusoidal line currents. The simulation results obtained have attested good dynamic performances and excellent robustness of this new control strategy.

References

1. Eskandari-Torbati, H., Khaburi, D.A., Eskandari-Torbati, V.: Virtual flux based direct power control (DPC) of three phase PWM rectifier using model predictive control (MPC) and space vector modulation (SVM). In: 5th Power Electronics, Drive Systems and Technologies Conference (PEDSTC), pp. 242–248 (2014)

2. Malinowski, M., Jasinski, M., Kazmierkowski, M.P.: Simple direct power control of three-phase PWM rectifier using space-vector modulation (DPC-SVM). IEEE Trans. Ind. Electron. **51**(2), 447–454 (2004)
3. Boudries, Z., Ziani, D.R., Sellami, M.: Direct power control of a PWM rectifier fed autonomous induction generator for wind energy applications. Energy Procedia **36**, 391–400 (2013)
4. Malinowski, M., Kazmierkowski, M.P., Trzynadlowski, A.: Review and comparative study of control techniques for three-phase PWM rectifiers. Math. Comput. Simul. **63**(3–5), 349–361 (2003)
5. Lalili, D., Mellit, A., Lourci, N., Medjahed, B., Berkouk, E.M.: Input output feedback linearization control and variable step size MPPT algorithm of a grid-connected photovoltaic inverter. Renew. Energy **36**(12), 3282–3291 (2011)
6. Zhang, Y., Li, Z., Zhang, Y., Xie, W., Piao, Z., Hu, C.: Performance improvement of direct power control of PWM rectifier calculation. IEEE Trans. Power Electron. **28**(7), 3428–3437 (2013)
7. Attaianese, C., Tomasso, G., Damiano, A., Marongiu, I., Perfetto, A.: Direct torque and flux control of induction motor drives. In: Proceedings of the International Conference on Power Electronics and Drive Systems, vol. 2, pp. 642–648 (1997)
8. Bakouri, A., Mahmoudi, H., Abbou, A., Moutchou, M.: Optimizing the wind power capture by using DTC technique based on artificial neural network for a DFIG variable speed wind turbine. In: IEEE 10th International Conference on Intelligent Systems: Theories and Applications (SITA), pp. 1–7 (2015)
9. Barara, M., Bennassar, A., Abbou, A., Akherraz, M., Bossoufi, B.: Advanced control of wind electric pumping system for isolated areas application. Int. J. Power Electron. Drive Syst. **4**(4), 567–577 (2014)
10. Bennassar, A., Abbou, A., Akherraz, M., Barara, M.: Sensorless backstepping control using an adaptive luenberger observer with three levels NPC inverter. WASET Int. J. Electr. Sci. Eng. **7**(8), 1102–1108 (2013)
11. Noguchi, T., Tomiki, H., Kondo, S., Takahashi, I.: Direct power control of PWM converter without power-source voltage sensors. IEEE Trans. Ind. Appl. **34**(3), 473–479 (1998)
12. Menghal, P.M., Laxmi, A.J.: Neural network based dynamic simulation of induction motor drive. In: 2013 International Conference on Power, Energy and Control (ICPEC), pp. 566–571 (2013)
13. Wu, X., Huang, L.: Direct torque control of three-level inverter using neural networks as switching vector selector. In: Industry Applications Conference, 2001. Thirty-Sixth IAS Annual Meeting. Conference Record of the 2001 IEEE, vol. 2, pp. 939–944 (2001)
14. Abdeslam, D.O., Wira, P., Merckle, J., Flieller, D., Chapuis, Y.-A.: A unified artificial neural network architecture for active power filters. IEEE Trans. Ind. Electron. **54**(1), 61–76 (2007)
15. Krishna, S.A., Abraham, L.: Boost converter based power factor correction for single phase rectifier using fuzzy logic control. In: 1st International Conference on Computational Systems and Communications (ICCSC), pp. 122–126 (2014)

Multicore Framework for Finding Frequent Item-Sets Using TDS

Sajid Gul Khawaja(✉), Amna Tehreem, M. Usman Akram, and Shoab Ahmed Khan

College of Electrical and Mechanical Engineering,
National University of Sciences and Technology, Rawalpindi, Pakistan
sajid.gul@ceme.nust.edu.pk, amnatehreeem@gmail.com, kshoab@yahoo.com

Abstract. Mining of frequent items from a dataset is a prime problem in the field of data mining. It plays a pivotal role in many of the data mining applications. In recent years, technological improvements have provided us with cheaper storage spaces capable of handling gigantic amount of digital data of human activities. This humongous data becomes a bottleneck for data analysts as mining algorithms often suffer from performance issues while running larger datasets or where number of items become very large. In this paper, we propose a novel tree based data structure (TDS) for saving itemsets as candidates for time effective application of Apriori property for pruning. The TD structure shows significant improvement in timing as compared to traditional structure. Furthermore, we propose a multicore framework for the processing of TDS based Apriori Algorithm. The framework is based on divide and conquer approach, where all cores work in parallel on their allocated subset of data. Each core shares their local results with other cores to get the global results leading to a collaborative working environment. The proposed framework is highly scalable which requires no change in the overall working of the algorithm. In order to thoroughly test the proposed framework, experimentation is performed using 4 benchmark datasets and its evaluation is carried out on the bases of number of cycles, execution time and comparative speedup. The results indicate that TDS is significantly faster and while working with multicore framework a direct relationship exists in speedup for all datsets with the number of working cores.

Keywords: Data mining · Apriori · Parallel processing · Multiple core · Accelerator

1 Introduction

Technological advancements in the recent past has helped in providing access to unfathomable amount of raw data, due to the availability of cheaper storage spaces and effective data sensing. Availability of such raw data has helped

A. Abraham et al. (eds.), *Proceedings of the 16th International Conference on Hybrid Intelligent Systems (HIS 2016)*, Advances in Intelligent Systems and Computing 552,
DOI 10.1007/978-3-319-52941-7_34

researchers in extracting knowledge (information) from data, which is called *Data Mining*. Data Mining techniques use the data to find unknown patterns (information) from raw data. This information is helpful in wide range of fields such as trend analysis [1], behavior of customers [2], disease diagnostics [3], bio-metrics [4] etc. which has enabled development of novel decision-support-systems.

Among various applications of data mining association rule mining is an important application [5]. Main concern of association rule is to find connections between different items, in a large database, on the bases of minimum confidence. Thus using association rule mining, a large data can easily be broken down into meaningful information. Frequent itemset mining mainly forms the backbone for association rule mining. Eclat, FP-Growth and Apriori are the algorithms mostly used in applications of frequent itemset mining. Among these perhaps Apriori is the most popular algorithm used to perform correlation-based Data Mining [6].

Apriori algorithm belongs to the category of candidate generation algorithms which initially construct candidate and then identifies list of frequent itemsets from them on the bases of minimum support (ξ). An anti-monotone property (namely Apriori [6]) forms the backbone of this class of algorithms, which is used to prune itemsets. It states that if any k-itemset is not frequent then all its $(k+1)$ super-set itemset will not be frequent. Generally, working of the algorithm can be described as: Generate $(k+1)$-itemsets in the $(k+1)^{th}$ pass using the itemsets generated in k^{th} pass, count the support i.e. frequency of these itemsets named candidates in the original database to find the frequent items. Overall the algorithm achieves good performance as it reduces the sizes of candidate itemsets.

In order to optimally perform the task of mining data from larger databases software based approaches offer very limited performance. This is supported by the fact that rate of data generation is exponential whereas performance of Data Mining applications has only increased by approx. 15% [7]. In this paper we propose a new tree based data structure (TDS) for saving candidate itemsets in an optimized manner. Furthermore, a multicore framework for mining frequent itemsets using Apriori algorithm is presented to accelerate the overall process of mining data.

The remaining paper is structured as follows: Sect. 2 discusses related work, in Sect. 3 proposed architecture is described, results and relevant discussion is carried out in Sect. 4. Finally, Sect. 5 concludes the paper.

2 Related Work

In this section, we discuss previous work done to enhance the performance of frequent itemset mining algorithms using Hardware (FPGA and GPU) Implementations.

Systolic Array architecture having hardware cells for speeding-up Apriori Algorithm was proposed in [8]. The architecture in this study was implemented on a Xilinx Virtex-II Pro 100 and provided performance improvement faster than

the state-of-the-art software implementations. The developed system, besides being scalable, introduced an efficient systolic injection method for automatic reporting of unwanted mid-array results to a controller minimizing the chances of collision or excessive stalling. Baker et. al. proposed a highly parallel custom architecture for data mining operations in [9]. This development, based on **bitmapped CAM**, utilized redundancy within the input data for efficient and simultaneous processing which significantly reduced the time and area required for executing the subset operations; essential to data mining.

A Hash-based and pipelined architecture for hardware enhanced association rule mining to overcome the problem of too many candidate itemsets and large databases was proposed in [10]. The pipelining method compared itemsets with the database and collected information to shrink the number of candidate itemsets as well as items in the database simultaneously. Another FPGA based hardware-enhanced framework for Apriori was proposed in [11]. The framework computes 1^{st} and 2^{nd} itemsets after a single scan of given datasets showing that the achieved throughput is much greater than traditional software based implementations.

In [10] bitmap data structure was presented to map the Apriori algorithm on a GPU and thus accelerating the counting process. Among other GPU based implementation [5] presented an alternate **static bitset** memory structure for saving the database. **Static bitset** memory structure allowed mapping of Apriori as a SIMD model on Nvidia Tesla $T10$ GPU providing $100\times$ speedup as compared to other algorithms on a CPU.

3 Proposed Model

In this paper, we present an altered data structure based on tree for improving the efficiency of mining frequent itemset. Furthermore, a multi-core architecture for parallel processing of Apriori Algorithm is proposed which makes use of the altered data structure. Thus, the overall paper has 2 parts namely modified data structure and Parallel framework for Apriori Algorithm using modified data structure.

3.1 Modified Data Structure

The general candidate generation class of algorithms perform well but studies have revealed that the repeated scans of database and larger number of candidates can make them highly expensive [12]. Thus, in this paper we first aim to present a new way, namely Tree Data Structure **(TDS)**, of saving the candidate itemsets which can expedite the process of searching and checking the Apriori property. Apriori property finds $k-1$ subsets of k itemsets and checks if it is present in the frequent patterns found in the previous iteration or not. If all the subsets are found then the k-itemset becomes a candidate set. Checking Apriori property and consequently finding candidate sets support are the time consuming parts of Apriori algorithm. This property needs to compare all

of these $k-1$ item subsets with the frequent patterns of length $k-1$. Let us say that $n*k$ represents frequent patterns of length k. In the worst case if a subset is not a frequent pattern, it will be compared against all $n*k-1$. With each of these frequent patterns it will take $(k-1)\times(k-1)$ comparisons. In order to find a mathematical expression stating the required number of cycles to find the candidate set let us assume transaction length to be 'n' and let NFP_i represent number of frequent patterns having length i. Equations 1 and 2 show the computations required to apply the Apriori property on 3^{rd} and 4^{th} itemset respectively.

$$3^{rd}-itemset=\frac{3!}{2!}\times 2\times 2\times NFP_2=12\times NFP_2 \tag{1}$$

$$4^{th}-itemset=\frac{4!}{3!}\times 3\times 3\times NFP_3=36\times NFP_3 \tag{2}$$

In a similar way, a generalized close form equation can be found which is shown in 3, simplifying this equation we are left with a second order equation (4) which shows that computational requirements increase exponentially.

$$n^{th}-itemset=\frac{n!}{(n-1)!}\times(n-1)\times(n-1)\times NFP_{n-1} \tag{3}$$

$$n^{th}-itemset=n\times(n-1)^2\times NFP_{n-1} \tag{4}$$

In this paper we have proposed a tree data structure, termed TDS, to save frequent patterns of the previous level. Main motivation in using TDS is to reduce the computational complexity of Apriori property. TDS saves all the itemsets C_i as L_i whose support is greater than the minimum support (ξ). The working of TDS is as: all the frequent patterns of length $k-1$, generated after k^{th} iteration are inserted in the tree. Taking root node as parent node, First item in each frequent patterns is made the child of that node. This newly added node is made the parent for next item and it is inserted in the tree, same process is repeated for all items. If children of the a node already contain an item then that child node is made parent on for insertion of next item.

Now while checking Apriori property using TDS all the items forming a pair with any item are found in nodes of the tree, thus reducing its computational complexity because the maximum number of comparisons for each subset with frequent patterns in the tree will be equal to the depth of tree which is $(k-1)$. Tree construction takes even less number of cycles then required for checking Apriori property for one set and after that there are fixed and very small number of cycles for checking Apriori property using tree. Equation 5 shows the closed form equation for cycle requirement using TDS.

$$n^{th}-itemset=\frac{n!}{(n-1)!}\times(n-1)=n\times n-1 \tag{5}$$

In order to understand the working how TDS saves frequent itemsets in the tree an example is shown in Fig. 1. In the example, a transactional dataset is processed using Apriori Algorithm keeping ξ as 2/9 while using TDS to save candidate set for 3^{rd} itemset.

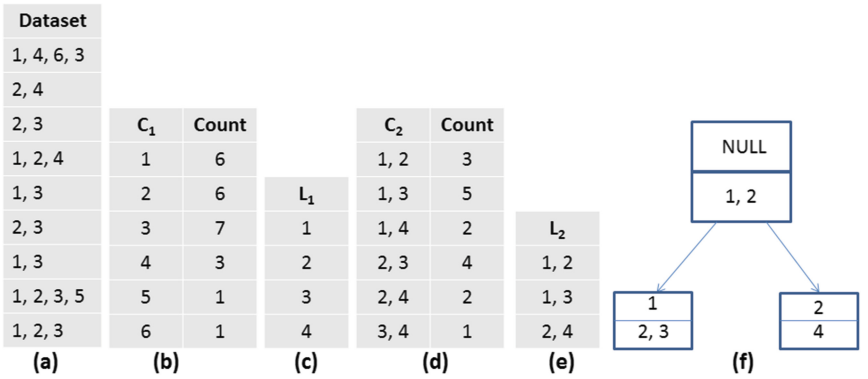


Fig. 1. Example showing generation of TDS (a) Dataset (b) 1^{st} candidate set and its support (c) 1^{st} itemset ($C_1 > \xi$) (d) 2^{nd} Candidate set and its support (e) 2^{nd} itemset ($C_2 > \xi$) and (f) 2^{nd} itemset saved using TDS

3.2 Proposed Parallel Framework

In this section, we discuss our proposed multi-core architecture for parallel implementation of Apriori Algorithm. The overall approach of our proposed hardware model is similar to ***Count Distribution*** algorithm presented in [13] as a software model for parallel computers. In our proposed architecture, the overall system is sub-divided into homogeneous cores where all the cores perform the same task on their allocated data. Furthermore, the cores are tightly coupled as after each iteration they collaborate with each other to complete an iteration. Figure 2 shows the top level view of the proposed structure and internal structure of a single core where each core has its own local memory.

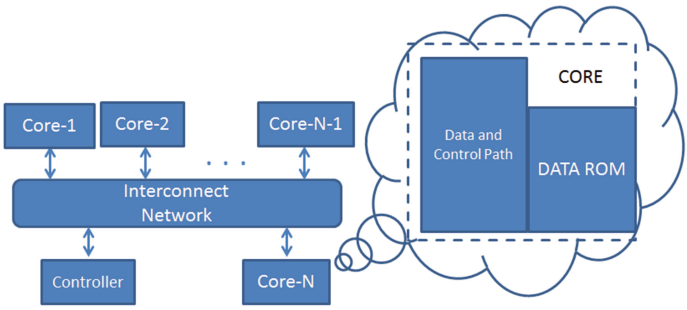


Fig. 2. (a) Top view of the proposed structure (b) Internal structure of a single core

As discussed earlier, the working of finding itemsets using Apriori algorithm can be roughly divided into following steps:

- Calculating Support Count for **Candidates**
- Trimming **Candidates** to get **Item-sets**
- Generating **Candidates** for next iteration

In the proposed architecture, these phases have been accelerated by splitting data into smaller chunks and then processing them in parallel. In order to understand the proposed model, let us state some basic notations; D represents the overall database; T denotes total transactions in the database; $I = \{i_1, i_2, i_3, \ldots, i_t\}$ represents total items in the transactions; C_i be the candidate list for i^{th} iteration; L_i be the list of item-sets after i^{th} iteration; $P = P_1, P_2, P_3, \ldots, P_N$ represents the processing cores with N being the total number of available cores; ${C_i}^p$ represents the candidate list in i^{th} iteration allocated to or by p^{th} processor.

Algorithm 1. Pseudocode representation of the proposed multicore framework for finding frequent itemsets

```
procedure PSEUDO CODE
    Input: database D
    Input: Max. Transaction Length
    Input: Total Number of unique items
    Output: frequent itemsets
    Initialization:
    Equally divide Data among N Cores → (D^p)
    Start:
    for all p ∈ P do                          ▷ Runs for 1st iteration only
        Generate local C_1^p for D^p
        C_1^p ← All Unique Items
        Calculate frequency of C_1^p
        Broadcast C_1^p in ascending order to all cores
        Update total C_1 with count received from other cores
        Prune candidates after applying minimum support (ξ)
        Find item-set L_1
    EndFor
    while L_{i-1} ≠ ∅ do                      ▷ For all other iterations
        for all p ∈ P do
            for all c ∈ C_i^p do
                Find candidate count using D^p
                Broadcast C_i^p and update count
                Find item-set L_{i+1} using ξ
                Generate C_i^p
        i := i + 1
```

Initially, the complete database is divided among N cores, where each core receives D^p a subset of D. In the first iteration, all the unique items in the database are kept as candidates at each core for easy synchronization. All cores start

with calculation of support for each candidate ${C_1}^p$ by searching their respective data D^p. After completion of support calculation, core $1 \rightarrow N$ respectively broadcast the support of candidates in an ascending manner. As any i^{th} core is broadcasting support of a candidate ${C_1}^i$, all other cores update the global support of that candidate C_1. Once all the cores broadcast support of all candidate the pruning of candidates on the bases of ξ takes place in parallel in all the cores to generate itemset L_1.

In the subsequent iterations all cores checks if L_1 is non-empty, in case of an empty set the process terminates. Otherwise $C_i \leftarrow L_{i-1}$, and all cores start working on the generated candidate list. In parallel all cores calculate support for C_i, broadcast and update support, prune C_i to get itemset L_i which are used as candidates for next iteration. A change in computation occurs for the generation of 3 and greater itemset. The L_i itemsets are saved using TDS and are further pruned by applying Apriori property which is applied in parallel in all the cores. These generated itemsets are processed unless an empty set is obtained. The simplified working of the proposed framework is shown in Algorithm 1.

4 Results and Discussion

The aim of this paper is to initially propose an altered storage structure for Apriori algorithm. The altered model is then modelled in a multi-core framework to provide timing efficiency by presenting a simplistic scalable architecture. In order to highlight the efficiency of the proposed data structure and eventually the overall multi-core framework multiple benchmark databases have been used for different minimum support. A total of 4 databases having real life data have been used for evaluation. Characteristics of the databases are summarized in Table 1; stating their average transaction length (ATL), maximum number of transactions (MT), percentage of dataset used to test, total number of unique items and minimum support (ξ). The chess and mushroom datasets are executed until they provide final itemsets whereas connect and pumsb dataset have been run till 4^{th} itemset calculation.

In the experiments, the performance evaluation has been carried out by computing the number of cycles, speedup against traditional Apriori algorithm and execution time. Initially the datasets have been processed using our proposed data structure and the conventional data structure. The effective speedup

Table 1. Summary of the datasets used in the experimentation

Dataset	ATL	MT	Unique items	% Utilized	Min. Support (ξ)
Chess	37	3196	75	100	25%
Connect	43	67557	129	10	25%
Mushroom	23	8124	119	100	15%
Pumsb	74	49046	2113	10	50%

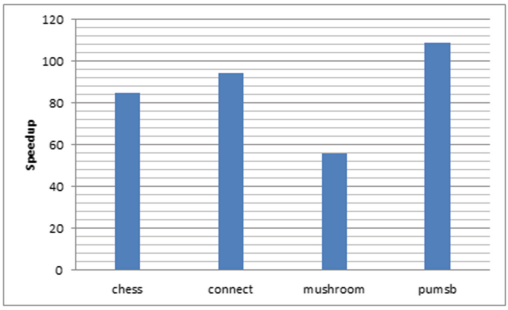


Fig. 3. Speedup comparison of proposed Tree based data structure (TDS) against conventional structure

Table 2. Execution time (in seconds) of conventional framework and the proposed Tree Data Structure

Type	Chess	Connect	Mushroom	Pumsb
Conventional	54.44762	30.95714286	55.149524	136.8381
TDS	0.643257	0.327595338	0.9873333	1.257524

achieved by the use of TDS in place of traditional data structure is shown in Fig. 3. Furthermore, Table 2 shows the comparative execution time for proposed and conventional models.

To evaluate the proposed multicore framework the datasets are processed using different multicore structures where $N = 2, 4, 8, 12$ *and* 16. The overall speedup comparison of the proposed multicore framework is done with a serialized model of TDS-Apriori algorithm. Figures 4 and 5 show the achieved speedup and number of cycles required to get the final results.

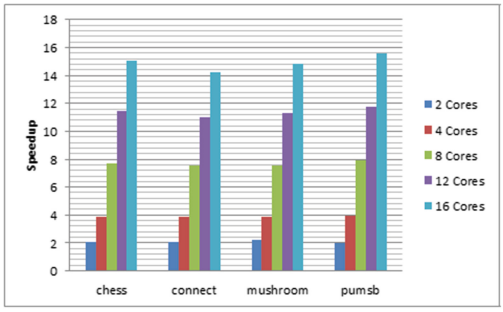


Fig. 4. Speedup comparison of Multicore TDS Apriori against TDS Apriori Algorithm

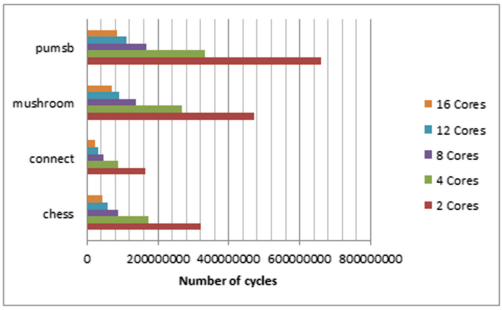


Fig. 5. Number of cycles required by our proposed Multicore TDS Apriori for different number of cores

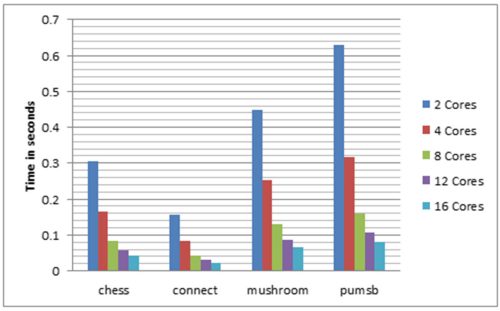


Fig. 6. Execution time required by our proposed Multicore TDS Apriori framework for different number of cores

In terms of execution time, Fig. 6 shows the execution time required in seconds by our proposed framework for different datasets. The results indicate that the performance of the proposed framework remains consistent on different datasets irrespective of the number of maximum available items. A linear improvement in timing is seen for all the datasets which advocates the use of proposed framework as a generic model for any dataset.

5 Conclusion

In this paper, a novel tree based data structure **(TDS)** has been presented to increase the timing efficiency of the algorithm while decreasing the searching time. The proposed TDS significantly decreases the number of cycles required to process a dataset while scanning for support of candidate itemsets. The Apriori algorithm using the proposed TDS has been further modelled using a multicore framework for the generation of frequent itemset. The proposed framework has the ability of being thoroughly scalable while returning a linear increase in the speedup and timing efficiency. The proposed TDS and the subsequent multicore

framework has been evaluated using 5 different benchmark datasets. The results show that a major improvement in timing performance is observed by the simple induction of TDS in place of tradition data structure. Furthermore, the multicore model also shows a directly proportional relationship between number of working cores (N) and the decrease in execution time.

References

1. Go, A., Bhayani, R., Huang, L.: Twitter sentiment classification using distant supervision. CS224N Project Report, Stanford, vol. 1, p. 12 (2009)
2. Mostafa, M.M.: More than words: social networks' text mining for consumer brand sentiments. Expert Syst. Appl. **40**(10), 4241–4251 (2013)
3. Foster, K.R., Koprowski, R., Skufca, J.D.: Machine learning, medical diagnosis, and biomedical engineering research-commentary. Biomed. Eng. Online **13**(1), 94 (2014)
4. Rao, C.R., Govindaraju, V.: Handbook of Statistics: Machine Learning: Theory and Applications, vol. 31. Newnes (2013)
5. Han, J., Kamber, M., Pei, J.: Data Mining: Concepts and Techniques. Elsevier (2011)
6. Agrawal, R., Srikant, R., et al.: Fast algorithms for mining association rules. In: Proceedings of 20th International Conference on Very Large Data Bases, VLDB, vol. 1215, pp. 487–499 (1994)
7. Thoni, D.W., Strey, A.: Novel strategies for hardware acceleration of frequent item-set mining with the Apriori algorithm. In: 2009 International Conference on Field Programmable Logic and Applications (2009)
8. Baker, Z.K., Prasanna, V.K.: Efficient hardware data mining with the Apriori algorithm on FPGAs. In: 13th Annual IEEE Symposium on Field-Programmable Custom Computing Machines (FCCM 2005), pp. 3–12. IEEE (2005)
9. Baker, Z.K., Prasanna, V.K.: An architecture for efficient hardware data mining using reconfigurable computing systems. In: 14th Annual IEEE Symposium on Field-Programmable Custom Computing Machines (FCCM 2006), pp. 67–75. IEEE (2006)
10. Fang, W., Lu, M., Xiao, X., He, B., Luo, Q.: Frequent itemset mining on graphics processors. In: Proceedings of the Fifth International Workshop on Data Management on New Hardware, pp. 34–42. ACM (2009)
11. Liu, W.-C., Liu, K.-H., Chen, M.-S.: Hardware enhanced mining for association rules. In: Ng, W.-K., Kitsuregawa, M., Li, J., Chang, K. (eds.) PAKDD 2006. LNCS (LNAI), vol. 3918, pp. 729–738. Springer, Heidelberg (2006). doi:10.1007/11731139_85
12. Han, J., Cheng, H., Xin, D., Yan, X.: Frequent pattern mining: current status and future directions. Data Min. Knowl. Disc. **15**(1), 55–86 (2007)
13. Agarwal, R., Shafer, J.C.: Parallel mining of association rules. IEEE Trans. Knowl. Data Eng. **6**, 962–969 (1996)

Toward Context-Aware SLA for Cloud Computing

Taher Labidi[1(✉)], Achraf Mtibaa[1], Walid Gaaloul[2], and Faiez Gargouri[3]

[1] Miracl Laboratory, National School of Electronic and Telecommunications, University of Sfax, Sfax, Tunisia
taherlabidi@gmail.com, achrafmtibaa@gmail.com

[2] Institut Mines-Telecom, Telecom SudParis, UMR CNRS Samovar, Evry, France
walid.gaaloul@telecom-sudparis.eu

[3] Miracl Laboratory, Higher Institute of Computing and Multimedia, University of Sfax, Sfax, Tunisia
faiezgargouri@gmail.com

Abstract. Service Level Agreements (SLA) represents the principal means to control the Quality of Service (QoS). In cloud computing, various advanced SLA strategies are used; some techniques among others try to avoid the SLA violations. Taking into account the cloud consumer contextual parameters is a promising way to predict and avoid costly SLA violations. Therefore, in this paper, we propose a context-aware system for the SLA in order to guarantee the QoS. We create a contextual ontology to introduce the semantic meaning of the cloud consumer contextual parameters. In addition, using reasoning techniques, we (1) predict SLA violations and we (2) choose and apply the corrective actions in autonomous manner. This allows getting a proactive adjustment of the cloud service execution and respecting the SLA parameters.

Keywords: Cloud computing · Service level agreement · Context-awareness · Ontology · Quality of Service

1 Introduction

Nowadays, efficient SLA management is a challenge in the cloud computing filed. SLA management proceeds in three steps namely the SLA establishment, the SLA validation and the SLA monitoring [11]. The first step consists of the definition and followed by the negotiation phase. These phases allow the specification of an SLA document. Then, the SLA validation step allows the prediction of violations and the interventions due to these violations. This phase must be performed throughout the service execution and in an automatic way to ensure that the QoS is always reliable. Finally, the SLA monitoring phase is achieved by detecting violations and verifying the reliability of the QoS. Thereby, the assurance of the QoS defined and controlled in the SLA represents a big challenge that remains unsolved. Therefore, advanced SLA enactment strategies must be

A. Abraham et al. (eds.), *Proceedings of the 16th International Conference on Hybrid Intelligent Systems (HIS 2016)*, Advances in Intelligent Systems and Computing 552,
DOI 10.1007/978-3-319-52941-7_35

ensured to duly react to failures and environmental changes. Parameters that cause the instability of the QoS must be deeply investigated in order to avoid costly SLA violations. Among these parameters, the dynamic change of cloud consumer context may negatively affect the QoS. Most of the works in literature ignore considering the consumer context. So, two cloud consumers with different context but similar SLA have equal importance for the service provider. To avoid this undesirable situation, the cloud consumer contextual parameters and their changes must be managed and planned in the SLA document in order to obtain a flexible SLA.

For this reason, we will study the different cloud consumer contextual parameters that may affect the QoS. In fact, these parameters varies widely, which can significantly influence how a service can be executed, and thus affect the actual performance of the service. To dynamically adapt the cloud service to the changes of the consumer context, it is necessary to study the context-awareness. Context-awareness is a technique for developing computing applications that are flexible, adaptable, and capable of acting autonomously on behalf of the users without any explicit user interaction [1]. Moreover, context awareness is the ability to sense and react to situation variations towards better operations. Thus, context critically matters in the completion of several tasks namely reasoning and interpretation. It can be seen as the set of parameters that can influence the services behavior. Taking the contextual parameters into account is so useful as it allows us to adapt cloud services depending on the context of the consumer and consequently avoid costly SLA violations. Hence, a good context information modeling formalism reduces the complexity of context-aware applications and improves their maintainability and innovativeness. Ontology based model is considered the most promising approach to provide a rich formalism for structuring contextual information [3]. We opt to improve the SLA by introducing contextual parameters to dynamically predict SLA violation and apply guarantee actions in order to ensure a reliable QoS. In this paper, we propose an ontology-based context-aware SLA approach to predict and avoid SLA violation. We define a contextual ontological model to semantically represent the cloud consumer contextual parameters. Moreover, we exploit the benefits of inference in ontology to predict SLA violations without any explicit user interaction. Following this violation prediction, we choose and apply corrective actions in autonomous manner. This allows getting a proactive adjustment of the cloud service execution to guarantee the QoS and respect the SLA parameters.

The remainder of this paper is organized as follows: In the Sect. 2, we will present the related work. Then, in Sect. 3, we intend to provide our ontology-based context-aware SLA approach. We detail our Cloud SLA Contextual Ontology (CSLAC'Onto) while showing its ability in modeling cloud consumer contextual parameters. Some inference rules are presented to demonstrate the power of our model in the violation prediction and its capacity to react automatically to adjust the resources consumption. We also give an instantiation example of our model and we show the automatic corrective actions one a violation is predicted. Finally, Sect. 4 will be devoted to the conclusion.

2 Related Works

The SLA is a legal document elaborated by the service provider. It is conceived to represent the best interests of the provider and not for the customer's requirements [15]. However, the cloud consumer requires that the QoS stay always reliable. Many efforts have been undertaken for this purpose. Emeakaroha et al. [6] present LoM2HiS framework to allow detecting violations and future threats in SLA. It can operate to prevent these threats by exploiting and managing cloud resources. Dastjerdi et al. [5] present architecture to automate the process of managing SLAs in the cloud. To semantically model and describe the SLAs monitoring capabilities, the WSMO ontology was used. This ontology built an inter-Cloud language that allows the semantic correspondence between the SLAs of different cloud layers. In addition, the effects of QoS dependencies between services that generate SLA violations were treated. To overcome these violations, the deployment of appropriate service monitoring and filtering report violation is performed using dependency knowledge. In addition, Garcia et al. [7] propose a new architecture headed by SLA for automatic provision, planning, allocation and dynamic resource management for Cloud. This architecture adapt the WS-agreement specification to the specific needs of cloud computing and allows the definition of QoS rules and automatic actions for arbitrary and corrective preventions. Morshedlou and Meybodi [13] present a new proactive resource allocation approach with aim of decreasing impact of SLA violations. They use two user's hidden characteristics, named willingness to pay for service and willingness to pay for certainty. New methods based on learning automaton for estimation of these characteristics are provided as well. Kouki [9] proposes an SLA-driven cloud elasticity management approach. He addresses the trade-offs between the benefit of SaaS vendor and customer's satisfaction. Thereby, his work aims to provide a solution to implement the elasticity of cloud computing to meet this compromise. Labidi et al. [12] propose an ontology-based Elasticity approach for the SLA in cloud computing. This approach applies automatically the corrective actions against the SLA violations that were already occurred. Introducing elasticity strategies in SLA and semantically represent them using ontology facilitate their comprehension. Moreover, using semantic inference rules allows choosing the best elasticity strategies to apply following the SLA violation. Advanced SLA techniques must provide a flexible approach to suit the dynamic nature of the cloud. However, we noted that there is a lack of work treated the SLA violations prevention in order to avoid them. Hence, taking into account the cloud consumer contextual parameters in SLA is a promising way to predict and avoid costly violations. Moreover, in order to duly react following the violations prediction, automatic corrective actions must be performed to guarantees the QoS.

3 Ontology-Based Context-Aware SLA Approach

In this section, we will present our approach for Context-Aware SLA in cloud computing to guarantee a reliable QoS for consumers and providers.

This approach is based on the context-awareness and is improved with semantic aspects. These improvements allow getting an autonomic context-aware system in the SLA where appropriate actions are taken to avoid costly SLA violations. Modeling consumer contextual parameters in cloud computing SLA is performed using an ontology. As a consequence, we take advantage of benefits in sharing knowledge and maintaining semantic information. Ontology contains different cloud consumer contextual parameters, their properties, their relationships and axioms to choose the most suitable corrective action (i.e. adding or removing resources) depending on the condition and situation of the other concepts. Moreover, we take another advantage of ontology which is the inference [2]. Inference is considered as one of the most powerful features of the semantic web. It makes each data item more valuable, because it can have an effect on the creation of new information. Each new piece of information has the capacity to add a great deal of new information via inference [8]. Hence, semantic inferences allow us to detect indirect correspondences that are not detectable by the different languages.

Our approach covers the context life cycle which is based mainly on the context modeling, context reasoning and context dissemination [14]. First, the cloud consumer contextual parameters will be modeled in the ontology and processed with SLA concepts presented in [10]. Next step, we create and execute inference rules to notify the consumers' context changes and to predict the SLA violations. Finally, we modify the service configuration by adding or removing resources once a violation is predicted. This should be automatically performed depending on the resources availability and the cloud consumer contextual parameters variation degree. These steps allow guaranteeing the QoS defined in the SLA and offers a reliable cloud service. Figure 1 illustrates the different steps of our approach.

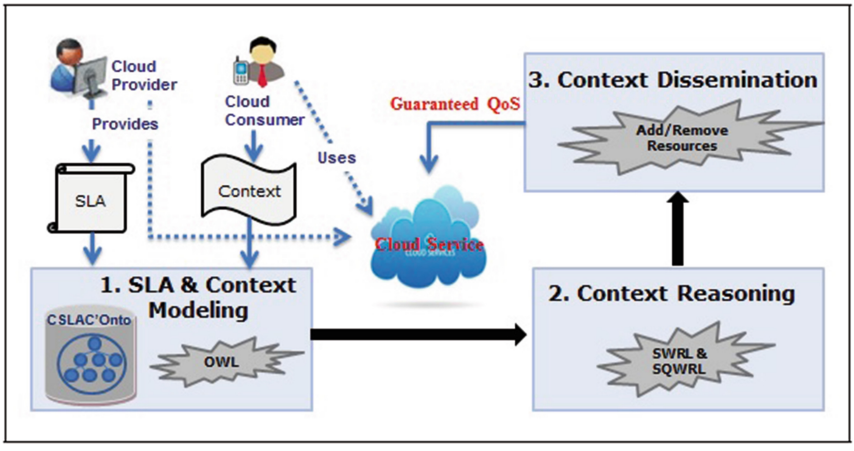


Fig. 1. Ontology-based context-aware SLA approach

Our approach supports the SLA validation step where the interventions due to consumer demand changes are made. It aims especially to prevent SLA violations and must be performed throughout the service execution in an automatic way to ensure a reliable QoS. It is composed of three main parts:

(i) SLA and Context Modeling: In this step, we create a Cloud SLA Contextual Ontology (CSLAC'Onto) containing the main cloud consumers' parameters. CSLAC'Onto is used to automatically predict SLA violations such as the service availability will exceed the defined sill. Ontological model allows modeling contextual parameters to be readable and comprehensible. In addition, it provides better coverage of the semantic relations between various concepts. CSLAC'Onto will be added to our generic Cloud SLA Ontology (CSLAOnto) presented in [10]. In this previous work, we have defined an ontological model containing the main concepts of the SLA document in the cloud computing.
(ii) Context Reasoning: Following the consumer contextual parameters variation, the QoS violation will be predicted in this step. In addition, the optimal service configuration has to be selected depending on the predicted violation degree. Thanks to reasoning techniques and inference rules defined in our ontology, we can select the best corrective action that guarantee the QoS.
(iii) Context Dissemination: In this step, the automatic implementation of the selected service configuration is performed. More precisely, in order to maintain QoS stable, we apply the best corrective action that has been chosen (i.e. add or remove resources: RAM, CPU, etc.). This allows the dynamic cloud service adaptation following the prediction of the SLA violation.

The goal of our ontology-based context-aware SLA approach is to avoid costly SLA violations by taking into account the cloud consumer context. Our approach is capable of acting autonomously on behalf of the users without any explicit user interaction. In addition, this approach adapts the cloud service execution to changing cloud consumer context. Ontology-based context-aware SLA approach has to perceive the situation of the cloud consumer and his resource utilization and consequently adapt the service configuration to that situation without an explicit demand from the provider or the consumer. This is beneficial for both cloud provider and consumer. Firstly, it assists the provider to implement corrective actions when violations are predicted. In this case, the provider can respond quickly and avoid the consequences of such violation in order to minimize penalties risk. In addition, it offers to the provider an advanced mechanism for controlling its available resources in such a way that they serve exactly the current customer demand in order to avoid excessive use of resources which maximizes its revenue. Secondly, it provides to the customer profitable services with acceptable QoS and reasonable usage cost. Moreover, it offers software, hardware and QoS requirement to the consumers in times. In this way, we guarantee a high profit provider while satisfying to the utmost the expectations of its customers.

3.1 SLA and Context Modeling

We use the Ontology Web Language (OWL)[1] to describe classes, constraints, and properties of our CSLAC'Onto. The OWL Language allows us to give semantic meaning to contextual parameters as well as combine and connect them in a manner understandable by machines. Moreover, we studied the different categorization schemes [14] and we highlight the need to combine them in order to complement their strengths and mitigate their weaknesses. Figure 2 presents the generic structure of our CSLAC'Onto which is linked to our Cloud SLA Ontology (CSLAOnto) [10]. CSLAOnto contains "SLACloud" concept as well as their "Terms" and "Parties" implicated. The parties are the Consumer, the Provider, the Auditor, the Broker and the Carrier. The cloud consumer is linked to his contextual parameters that cover the User "UserContext", Physical "PhysicalContext" and Computing "ComputingContext". These parameters influence on the variability of the customer requirements, the service usage and consequently on the QoS defined in the SLA.

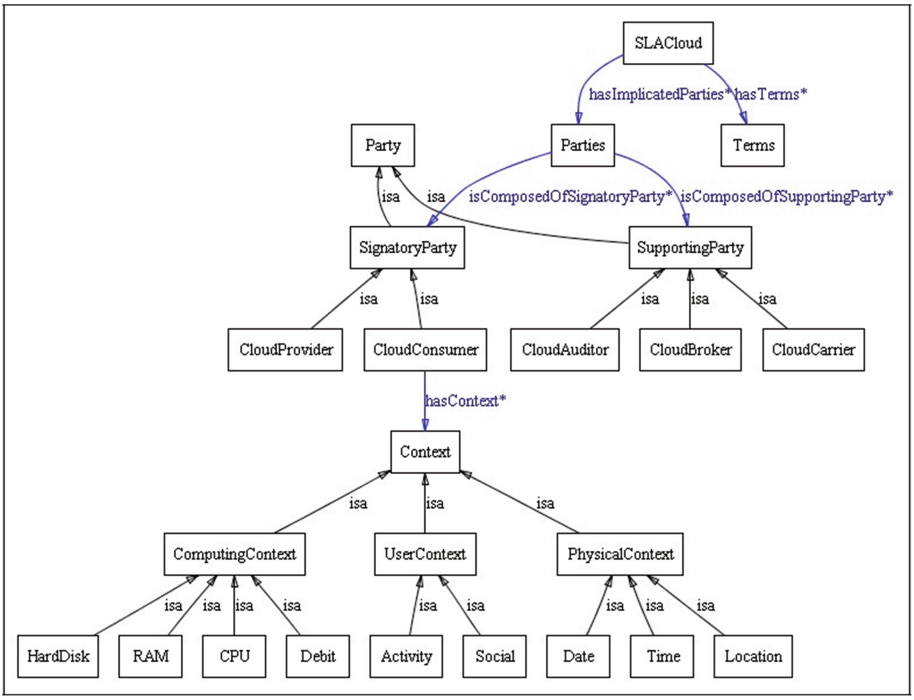


Fig. 2. Cloud SLA Contextual Ontology (CSLAC'Onto)

[1] http://www.w3.org/TR/owl-ref/.

- User Context: user context is composed of activity and social networking which give more details about the cloud consumer. Activity represents the cloud consumer function whereas the social networking includes the whole social networking that the consumer possesses.
- Physical Context: physical context represents the time, the date and the location concepts. They encompass the cloud consumer environment.
- Computing Context: computing context is composed of CPU, Debit, RAM and Hard Disk. These contextual parameters are very important since they provide knowledge about the device from which the consumer is connected to use the cloud service.

3.2 Context Reasoning

To ensure a good exploitation of this ontology, a reasoning step is essential. We have defined some inference rules using the Semantic Web Rule Language (SWRL)[2]. A SWRL rule has a semantic meaning by expressing the relation between an Antecedent and a Consequent in order to infer or deduce new knowledge from a set of implicit data. An Antecedent represents a conjunction of one or more atom aiming together to define the conditions that must be met. Whereas, the Consequent specifies the fact that may be resulted in case of fulfillment of the conditions defined in Antecedent [4]. SWRL rules can contain also SWRL built-ins (e.g. actions:ChooseAppropriateAction) or SQWRL queries (e.g. sqwrl:select). SWRL built-ins are user-defined predicates, including basic mathematical operators and functions for string manipulations. SQWRL queries define a set of operators used to exploit the knowledge inferred by SWRL rules. We have made some simulations in "Protege"[3] to ensure the proper functioning of our defined inference rules. Thus, we have populated our ontology with individuals, defined object properties and data type properties related to those individuals and launched the Jess[4] inference engine.

Table 1 shows some inference rules in SWRL that are defined to choose the corrective actions which preserve the QoS. Using contextual parameters, we can predict a violation and deduce the actions that will be performed. For example, following the first rule, we can deduce that we have to add CPU resources to

Table 1. Inference rules of the CSLAC'Onto

Rule no	Domain rules
1	$CloudConsumer(?x) \wedge hasContext(?x, ?y) \wedge ComputingContext(?y) \wedge CPUValue(?y, ?z) \wedge swrlb:greaterThan(?z, Sill) \wedge actions:ChooseAppropriateAction(?AppropriateAction,$ "$AddCPU$", $?z, ?CPUtoAdd) \rightarrow sqwrl:select(?AppropriateAction)$
2	$CloudConsumer(?x) \wedge hasContext(?x, ?y) \wedge ComputingContext(?y) \wedge RAMValue(?y, ?z) \wedge swrlb:lessThan(?z, Sill) \wedge actions:ChooseAppropriateAction(?AppropriateAction,$ "$RemoveRAM$", $?z, ?RAMtoRemove) \rightarrow sqwrl:select(?AppropriateAction)$

[2] http://www.w3.org/Submission/SWRL/.
[3] http://protege.stanford.edu/.
[4] http://www.jessrules.com.

the Virtual Machine (VM) hosting the cloud service. To do this, we first check if the CPU value of the device from which the cloud consumer is connected is greater than the defined Sill. The first two lines of this rule are designed for this purpose. Next, we calculate the required CPU to add for this service. This is performed with our defined built-in "action: ChooseAppropriateAction". It allows choosing the most appropriate action among those defined in our ontological model according to the degree of cloud consumer context variation. Finally, the "sqwrl:select (?AppropriateAction)" returns the action chosen by our built-in that will be applied to the service. The second rule allows deducing that we have to remove RAM resources from the VM hosting the cloud service. The built-in "swrlb: lessThan" is designed to compare the RAM value of the device from which the cloud consumer is connected with the defined sill. The result of this rule provides the necessary information to implement management action for this situation. The built-in "action: ChooseAppropriateAction" calculate the RAM value to remove while "sqwrl:select (?AppropriateAction)" returns the amount of the RAM to be deleted. In the same manner, we can deduce that we have to add or remove other computing resources. In addition, we can combine these rules to automatically choose appropriate actions following various contextual parameters changes appeared in the same time. Whereas, in the same manner, we can use the other contextual parameters (user and physical context) to define the action to be taken in order to predict the SLA violation.

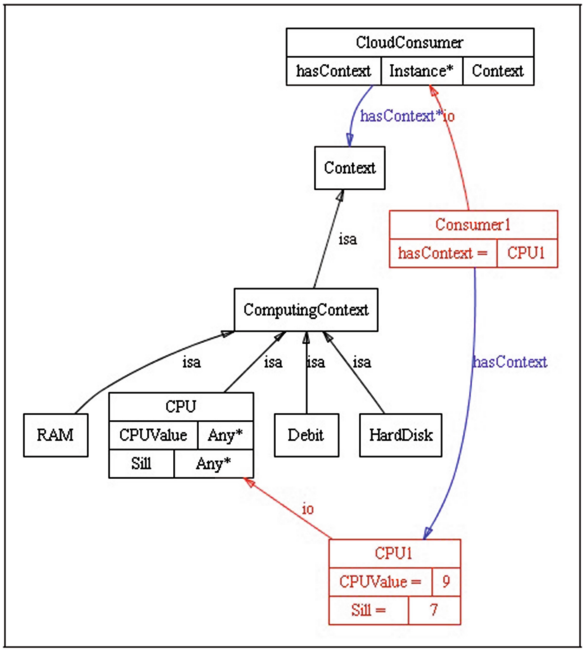


Fig. 3. CSLAC'Onto instantiation example

3.3 Context Dissemination

To validate our ontological model and its defined rules, we have created a CSLAC'Onto model instance. In order to demonstrate the performance of the first rule, we instantiated all their concepts. Figure 3 shows an example of this instantiation.

In this example, the cloud consumer context has been changed. In particular, the CPU value of the device from which the cloud consumer is connected exceed the sill defined. So it is required to add CPU resources for this service. When executing the first rule with Jess inference engine, our built-in "Action: ChooseAppropriateAction" allows choosing the most appropriate action among those defined in our ontological model. In our example, the action chosen is to add 2 CPU as described in Fig. 4 visualized through SWRLtab plugin. This action is based on elasticity strategies and take into account the cloud resources availability as explained in [12].

Fig. 4. The result of rule 1 execution

4 Conclusion

In the context of cloud computing and in order to guarantees the QoS specified in the SLA, it is imperative to study the factors that negatively influence the QoS. These factors cause a variation in the cloud service consumption and consequently in the QoS measures such as availability and response time. Among these factors, we highlight the cloud consumer contextual parameters that have not been so far considered in the SLA document. In this paper, we present an ontology-based context-aware SLA approach in cloud computing. Using the benefits of representation and inference of ontology, we can semantically model the contextual parameters and choose the appropriate corrective actions to be used following an SLA violation prediction. Our context-aware system allows reacting to consumer context variations towards better operations. We tested our approach with an example using inference rules in ontology to show the efficiency and effectiveness of the proposed approach. The result of this work is the basic tool for reliable QoS and ameliorated SLA while satisfying cloud consumers and providers' requirements. In the ongoing work, we are testing our approach in a real cloud computing platform.

References

1. Abowd, G.D., Dey, A.K., Brown, P.J., Davies, N., Smith, M., Steggles, P.: Towards a Better Understanding of Context and Context-Awareness. In: Gellersen, H.-W. (ed.) HUC 1999. LNCS, vol. 1707, pp. 304–307. Springer, Heidelberg (1999). doi:10.1007/3-540-48157-5_29
2. Babu, A., Sivakumar, R.: Development of ontology based middleware for context awareness in ambient intelligence. In: 2014 14th International Conference on Hybrid Intelligent Systems (HIS), pp. 219–224,., December 2014
3. Bettini, C., Brdiczka, O., Henricksen, K., Indulska, J., Nicklas, D., Ranganathan, A., Riboni, D.: A survey of context modelling and reasoning techniques. Pervasive Mob. Comput. **6**(2), 161–180 (2010)
4. Brabra, H., Mtibaa, A., Sliman, L., Gaaloul, W., Gargouri, F.: Semantic web technologies in cloud computing: a systematic literature review. In: 2016 IEEE International Conference on Services Computing (SCC), pp. 744–751. June 2016
5. Dastjerdi, A.V., Tabatabaei, S.G.H., Buyya, R.: A dependency-awareontology-based approach for deploying service level agreement monitoring services in cloud. Softw. Pract. Exp. **42**(4), 501–518 (2012). doi:10.1002/spe.1104
6. Emeakaroha, V.C., Brandic, I., Maurer, M., Dustdar, S.: Low level metrics to high level SLAs-LoM2HiS framework: bridging the gap between monitored metrics and SLA parameters in cloud environments. In: 2010 International Conference on High Performance Computing and Simulation (HPCS), pp. 48–54. June 2010
7. Garcia, A.G., Espert, I.B., Garcia, V.H.: SLA-driven dynamic cloud resource management. Future Gener. Comput. Syst. 31, 1–11 (2014). Special Section: Advances in Computer Supported Collaboration: Systems and Technologies. http://www.sciencedirect.com/science/article/pii/S0167739X1300215X
8. Hebeler, J., Fisher, M., Blace, R., Perez-Lopez, A.: Semantic Web Programming. Wiley, Indianapolis (2009)
9. Kouki, Y.: SLA-driven cloud elasticity management approach. Theses, Ecole des Mines de Nantes, https://tel.archives-ouvertes.fr/tel-00919900
10. Labidi, T., Mtibaa, A., Brabra, H.: CSLAOnto: a comprehensive ontological SLA model in cloud computing. J. Data Semant. **5**(3), 179–193 (2016). doi:10.1007/s13740-016-0070-7
11. Labidi, T., Mtibaa, A., Gargouri, F.: Ontology-Based Context-Aware SLA Management for Cloud Computing. In: Ait Ameur, Y., Bellatreche, L., Papadopoulos, G.A. (eds.) MEDI 2014. LNCS, vol. 8748, pp. 193–208. Springer, Heidelberg (2014). doi:10.1007/978-3-319-11587-0_19
12. Labidi, T., Mtibaa, A., Gargouri, F.: SLA Ontology-Based Elasticity in Cloud Computing. In: Morzy, T., Valduriez, P., Bellatreche, L. (eds.) ADBIS 2015. CCIS, vol. 539, pp. 145–152. Springer, Heidelberg (2015). doi:10.1007/978-3-319-23201-0_17
13. Morshedlou, H., Meybodi, M.R.: Decreasing impact of SLA violations: a proactive resource allocation approach for cloud computing environments. IEEE Trans. Cloud Comput. **2**(2), 156–167 (2014)
14. Perera, C., Zaslavsky, A.B., Christen, P., Georgakopoulos, D.: Context aware computing for the Internet of Things: a survey. CoRR abs/1305.0982 (2013)
15. Wieder, P., Butler, J.M., Theilmann, W., Yahyapour, R. (eds.): Service Level Agreements for Cloud Computing. Springer, New York (2011)

An Adapted Entity Summarization Service for an Enhancement Video System

Olfa Ben Said(✉), Ali Wali, and Adel M. Alimi

REGIM: Research Groups on Intelligent Machines, National Engineering School of Sfax (ENIS), University of Sfax, BP 1173, 3038 Sfax, Tunisia
{olfa.bensaid,ali.wali,adel.alimi}@ieee.org

Abstract. Knowledge bases' field has known a huge development these last years. It represents the interest of several types of systems, such as intelligent systems, search engines and several other applications. In some applications, this type of data becomes unsuitable because of the large amount of information it contains. From this problem has emerged the appearance of entity summarization methods.

In this work, we adopt a summarization service API and we adapt it to our previous work [5] for the enhancement of video with the knowledge base DBpedia. We aim by this work to enhance information received by the video's viewer with extra-data and present it in a concise and comprehensive form to be more comfortable and representable to the user.

Keywords: Entity summarization · Enhancing video · Interlinking · Open linked data · DBpedia

1 Introduction

The amount of Linked Data published on the Web is increasingly growing. These resulting entities contain descriptions including thousands of Triples. This large amount of information makes difficulties for human to comprehend the data contained in the entity, except if a selection of most relevant facts is performed. This task is called entity summarization.

In order to interlink video contents with entities from DBpedia, we have to present the entity in a concise and comprehensive form. Due to the specificity of video interfaces, the concise form will reflect on the quality of information dissemination and will create a flexible interactivity with the user.

A similar work is presented in ELES [7], where authors propose a combination between DBpedia Spotlight and an entity summarization interface. The purpose of the system is to analyse text and links fragments to entities of the knowledge base DBpedia. Entities are then presented in a summarized way. Xlime[1] is a semantic search prototype which propose different service such as keyword

[1] http://xlime.eu/.

A. Abraham et al. (eds.), *Proceedings of the 16th International Conference on Hybrid Intelligent Systems (HIS 2016)*, Advances in Intelligent Systems and Computing 552,
DOI 10.1007/978-3-319-52941-7_36

search, entity search and live updates. With this service the result of search is presented with a summarization service, in order to enable the users to quickly assimilate important information about entities and for improved navigation. Aemoo [1] is an exploratory search system which supports EKP(Encyclopedic Knowledge Patterns)-driven knowledge exploration and integrates data coming from heterogeneous resources. The system proposes a summary description according to the entity type.

Most of entity summarization methods are based on the ranking of facts present with the entity. Methods differ according to application needs. The aim of this service is to present a knowledge database entity in a concise and comprehensive form due to the large number of data existing into an entity. In this paper, we discuss some entity summarization methods and we present the adopted method for our system.

In Sect. 2, we define briefly the Enhancing system and the main idea of this last. A brief description of the interlinking sub-system is presented with its different components in Sect. 3. In Sect. 4, we dealt with the adopted Entity Summarization service and we present the result of its adaptation with the enhancement system. Finally a conclusion is given in Sect. 5.

2 Enhancing System

Large amounts of videos are published every day into the web affecting different topics. Our idea consists of enriching the main topics of a video with extra information from DBpedia. Figure 1 illustrates this idea. We aim, by this work, to engage videos viewers in an intellectual context and makes video an entercation (entertainment+education)[2] environment. Moreover, we aim to interlink videos

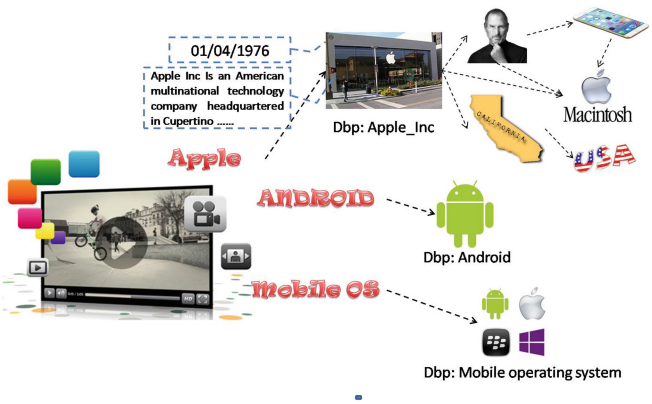


Fig. 1. Proposed system

[2] This term is used in [17] also in the web, having the mean of entertainment that educates.

specifically with a Knowledge Base connected to Linked Data in order to enrich videos and make them connected to a variety of sources and data. Starting from an input video, we must ensure connection from the video towards the knowledge Base DBpedia. In [2] and [3], we presented the main idea of the system and the proposed architecture with its different components.

3 Interliniking Sub-System

In this section, we focus on the interlinking process, which aims to link videos description with DBpedia Dataset. As presented in Fig. 2, the interlinking subsystem is composed of four main steps: Candidate Lookup service, Documents Preprocessing, context-based disambiguation and Entity summarization Service.

In the Lookup step, the system builds his space candidate resources according to the searched keyword. Document preprocessing consists of modeling the list of candidates and applying a list of pretreatment steps. The disambiguation consists of applying ranking function to choose the most relevant candidate according to query's vector and candidates' vectors. The process of the last three components is published in [4,5], where we focus on defining the adopted disambiguation algorithms. In the summarization service, we aim to filter only the main proprieties for the selected entity. Otherwise, this service aims to summarize selected entities and presents main information concisely. The aim of this paper is to present Entity Summarization and to define how it is adapted to our Interlinking system.

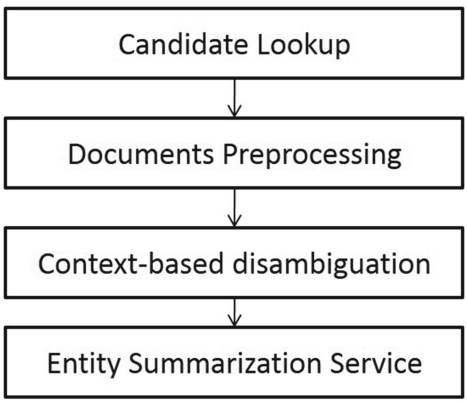


Fig. 2. System components

4 Entity Summarization Adapted to Interlinking System

Entity that enhances video contents contains different information in the form of RDF triplet. Resource's information (facts) varies from one resource to another,

depending on the entity membership and the popularity of the resource. For example the entity DBp:BobMarley, which belongs to Artist's class, contains birthDate, birthName, BirthPlace, Religion, musicalBand of, producer of, etc. Whereas, the entity dbp:Apple_Inc, which belongs to Company class, contains different facts, such as foundedBy, foundingDate, locationCity, service, product, industry, etc. In addition, two sources having the same class can differ in representing its facts, if one resource is connected to several external links due to its popularity. Most of Entity Summarization work focuses on some criteria in the selection of facts, such as popularity, relatedness, informativeness, etc.

The aim of this task is to present the entity selected for the video in a concise form instead of presenting all the entity. Due to the specificity of our system, this task is primordial and will improve the visualization of our application.

4.1 Background

Knowledge bases field has known a huge development these last years. Indeed, they are increasingly developed and highly used by intelligent systems, search engines and several other applications. With the wealth of these data, comes the problem of the navigation into this latter and the extraction of important facts and information. Henceforth appeared the entity summarization methods. The big challenges in this field of search, is to decide which information is important and according to which criteria we have to extract these information.

In some cases and for some applications, the large amount of information which contain an entity, becomes embarrassing. That is why; it would be very valuable to have systems that extract only limited information from such a graph. This problem is called Graphical Entity Summarization (GES). As regards GES problem, an entity is specified by a user or generated by a system (e.g. following an interlinking stage, as it is in our case), in order to give to the system the ability to summarize facts of the entity.

In the last years, entity summarization's field has gained particular attention by both industry and research. On one hand, the commercial approaches have the propriety of being very specific to their individual settings and rely on large amounts of background information. As such, these approaches can neither be generally applied nor reproduced. As to approaches from scientific field, they are more generic and generally applicable. In the next, a list of criteria is extracted that can make a good entity summarization [10]. To develop an entity summarization system, we must ensure the maximum of these criteria. In the bibliographic studies presented in the next section, we rely on these criteria for the choice of the adopted method.

Conciseness

A good summary should have a concise form and not very long representation. Conciseness can be declared in two forms, with a given percentage of the facts that will be presented or with an upper limit fixed either by the developer or by the user. The second method is better than the first one, due to the fact that the amount of facts contained in an entity varies from one entity to another. For example 20% can represent 1000 facts for an entity while only 10 for other entity.

Relevance
Relevance is the topological proximity of an arc to the query-entity in the semantic knowledge graph. I.e. the arcs that are less number of hops from the summarized node represent facts that are more "relevant" to this node.

Diversity
It is more convenient if the summary presents a diverse choice of predicates (type of relation). It becomes boring if we get a repetitive list of the same relation. For example for the example of William Shakespeare, for the property "author of" we can get "First Faulio", "Mac Haumor", "Shakespear: The animated Tales", etc. Diversification aims to avoid this problem to provide a more complete entity's overview.

Importance/Informativeness
Importance is based on the arcs' weights that represent importance of the fact. Different methods are proposed for this metric, the most used one is the PageRank algorithm and other variants such a Random surfer.

Popularity
For each entity, there are some types which convey valuable information. Popularity is based on the statistical frequency of arc label in the topological neighbourhood of the summarised node in the graph. Example the type *English Poet* is very important if we talk about *William Shakespeare*, while it is less important if we talk about *People of the Tudor period*.

4.2 Entity Summarization Overview

The Entity Summarization field, that has recently been studied, aims at summarising information around a single node in a graph or an entity. The first proposition was presented by [9], who proposes an algorithm called PRECIS. PRECIS is a diversity-oblivious approach, which try to simply selects the K edges closest to the entity. The weight's arc of this method is based on the "witness count" value. It is computed during the knowledge harvesting procedure and reflects the number of times the fact represented by the arc was found in the documents of the processed base text corpus. An experiment presented in [10], shows that for the query 'Tom cruise', the summarization algorithm extract only edges 'actedIn'. This result is clearly not optimal. DIVERSUM [18] is a diversity-aware approach. Unlike to PRECIS, DIVERSUM doesn't suffer from the redundancy problem. It tries to produce a diversified output by avoiding arc label redundancy. DIVERSUM summary is constructed by adding the edge having the shortest distance to the entity and the most frequently used property in the data. Especially, in order to improve its diversity, DIVERSUM does not generate features sharing a common propriety. An evaluation presented in [10], shows that the DIVERSUM algorithm was favored over the PRECIS algorithm.

A similar proposition called RELIN [11], uses a random surfer model on a graph to rank features that characterize the entity. The main idea of RELIN is to combine textual notions of informativeness and relatedness for the ranking

Table 1. Overview of summarization systems

	Relevance	Importance	Popularity	Diversity
PRECIS [9]	√	√		
RELIN [11]	√	√		
DIVERSUM [18]	√	√	√	√
SUMMA [6]	√	√	√	

of features. As a major effect, it results to a quick identification of the entity. RELIN does not look at diversity.

[13] also considers that the importance of entity's feature is related to the number of relation with nearest neighbours. So, it exploits usage data (such as ratings) to identify the most shared proprieties with the k neighbours. Like PRECIS and RELIN, this method does not consider diversity.

A recent work to Thalhamer called SUMMA was proposed in [6]. It consists of an API definition that presents an entity summarisation service with DBpedia, which combines diverse summarisation approaches. To develop this API, Thalhamer proposes The SUMMA vocabulary, which offers different parameters that represent a summary and a RESTful Web Service. We adopt this method in our enhancement video application and adapt it to the needs of our system. The method adopted by SUMMARUM ranks objects (ongoing links) based on the number of Wikipedia's incoming links and makes it available with vRank vocabulary. In an ulterior version [8], vRank was updated by combining two measures: one that accounts for the importance of the connected resource (PageRank) and one that accounts for the strength of the connection (Backlink). Table 1 outlines some summarization propositions presented above and the different criteria respected by each technique. All methods present already summaries of entities in a top-K manner to ensure Conciseness criteria.

The entity summarization is also a part of some search engines, such as Google (GKG) [14], yahoo Knowledge [15] and Microsoft Bing Satori/Snapshots [16]. Google Knowledge graph (GKG) aims at enriching results of the user query with summarized information about disambiguated entities. Google bases on its summarization on contexts about data items (e.g. abstract from Wikipedia, birthdate, population, etc.) for some facts. In addition, some proprieties are always present, such as abstract, pictures, entity names. GKG supports RDF to enable the adaptation of summarisation according to the preferred language. Bing extracts features similar to GKG (grouping, special property, context and multi-languages). As to Yahoo Knowledge, it displays his service only for persons and movies. Summarisation in Yahoo is not available for all languages.

4.3 Adopted Method

After a literature overview, we conclude that SUMMA API [6] is the most suitable to our needs. It includes different criteria. SUMMA ensure relevance by

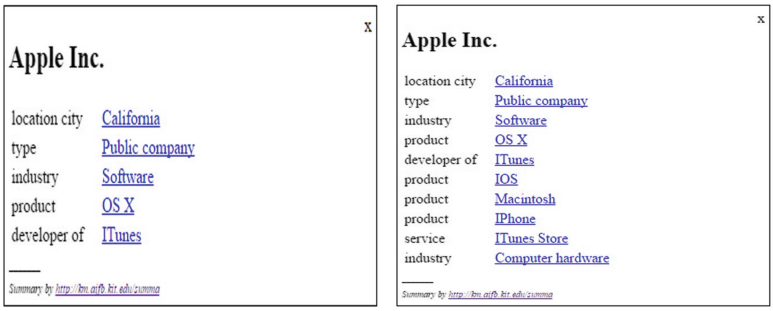


Fig. 3. Result of SUMMA summarization system of the entity dbp:Apple_Inc., with a topK = 5 and topK = 10

including the property maxHops[3]; In fact, relevance is ensured when setting maxHops to 1. A popularity importance-based approach is also proposed (PageRank & Backlink) [8], which computes the Rank Scores for each statement, using the vRank[4] vocabulary (Vocabulary for Ranking). As to conciseness, SUMMA defines a property called topK, which defines the number of statements that will be returned.

Other property presented in SUMMA is the possibility of defining summarize with the chosen languages. Likewise, SUMMA is adopted to DBpedia entity and provides an open-source reference implementation. After configuring all the cited parameters; SUMMA represents the most adequate service that could be adopted. Therefore, we adopt this service and adapt it to our system after making the appropriate changes.

This approach consists of two main components: SUMMA Vocabulary and the RESTful web service. SUMMA API server adopt these notions to make a uniform interface for entity summarization that meet the standard of the Linked Data model. Figure 3 represents the result of SUMMA summarization system of the entity dbp:Apple_Inc., presented in English language with a topK = 5 and topK = 10.

Our proposition consists of adapting the SUMMA API to our interlinking system. The adaptation consists of: first adding some specific proprieties to the original summarization service, second combining the summarization service with the interlinking system. In fact after the decision of selecting the appropriate entity by the system, the selected entity will be presented with the enhanced SUMMA API. In other word, this combination aims to present the selected entity in a concise and comprehensive form and to enhance the video annotation's with data in a light way that suits to the application's need.

[3] maxHops is a proprety which represents the maximum number of hops that the interface can represent.

[4] vRank aims to model ranking information within datasets. vRank: http://purl.org/voc/vrank#.

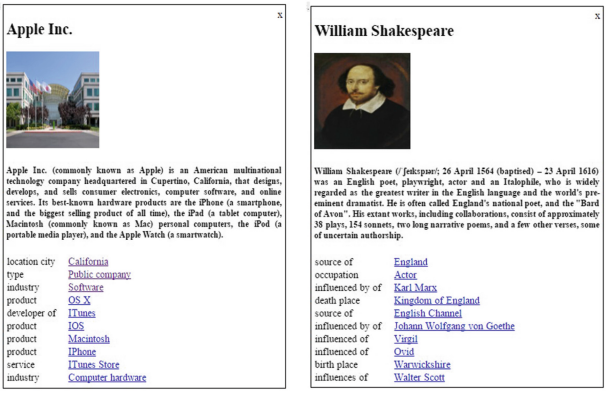


Fig. 4. A screen-shot of the SUMMA Summarization system after modification with dbp:Apple_Inc. and dbp:William_Shakespeare

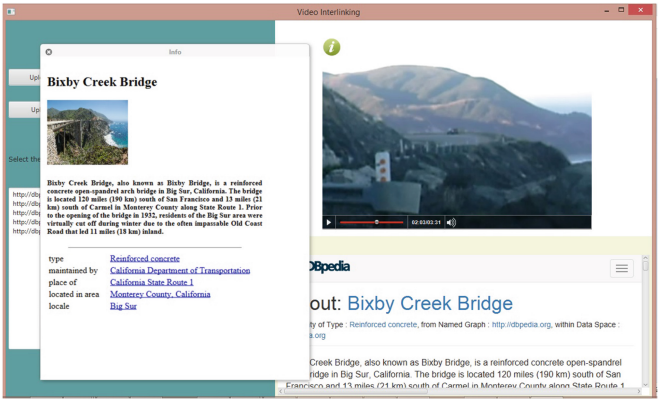


Fig. 5. Screen-shot of a video about "Bixby Bridge" linked to DBpedia entity with Summarization system

As we presented above, the SUMMA service presents a good summarization system, which responds to different criteria. Nevertheless, there is still a lack of some information. As Goggle or Yahoo summarization system, we catch up the idea of "specific properties". We make the necessary changes to the existing service, and we add the abstract and a picture that describes the entity to the old SUMMA interface. We get the examples present in Fig. 4

After adapting the summarization system to our application by adding the necessary information and configuring the required parameters (language, MAx-Hops and TopK), we make combination between the two application and we get a screen shot of the output result in Fig. 5. In this figure, we present an example of interlinking video to an entity and representing it in a summarized way.

With the summarization service, the user can quickly identify the desired entity and the interface is more suitable to the type of application.

5 Conclusions

On one hand, the nature of our enrichment system requires a specific presentation of the added content. In fact, the way of representing the enriched video can play a major role to attract the user's attention and to deliver the information properly. On the other hand, RDF DBpedia entity contains a big number of facts, which become unsuitable to our interlinking system. For this reason, we propose to present the linked entity in a user-friendly way by applying an entity summarization system called SUMMA and adapt it to our system's need. The result was a system that propose a DBpedia entity for each segment of the video, to enhance the video content's. The linked entity is represented in a concise and suitable form. In the future works, we aim to make some improvements to our prototype. First we aim to add a navigational elements to support contextual and semantic navigation. We aim also to propose a user interaction service which allow the user to interact and choose their satisfaction about the proposed interlinked data.

Acknowledgments. The authors would acknowledge the financial support of this work by grants from General Direction of Scientific Research (DRGST), Tunisia, under the ARUB program.

References

1. Nuzzolese, A., Presutti, V., Gangemi, A., Peroni, S., Ciancarini, P.: Aemoo: Linked data exploration based on knowledge patterns. Semant. Web **8**, 1–26 (2016)
2. Ben Said, O., Karray, H., Haqiq, A., Alimi Alimi, M.: T-learning and interactive television for edutainment. In: International Conference on Engineering Education and Research (ICEER), Marroco, pp. 258–264 (2013)
3. Ben Said, O., Wali, A., Alimi, A.M.: A new System for TV program contents improvement using a semantic matching technique. In: 13th International Conference on Hybrid intelligent System (HIS), Yesmine Hammamet, Tunisia (2013)
4. Ben Said, O., Wali, A., Alimi, A.M.: Interlinking video programs with Linked Open Data. In: 15th International Conference on Intelligent Systems Design and Applications (ISDA), Marrakech, Morocco, pp. 246–250 (2015)
5. Ben Said, O., Wali, A., Alimi, A.M., Interlinking video with DBpedia using knowledge-based Word Sense Disambiguation algorithms. J. Inf. Assur. Secur. (JIAS) **6**(11), 322–330 (2016)
6. Thalhammer, A., Stadtmüller, S.: SUMMA: a common API for linked data entity summaries. In: Cimiano, P., Frasincar, F., Houben, G.-J., Schwabe, D. (eds.) ICWE 2015. LNCS, vol. 9114, pp. 430–446. Springer, Heidelberg (2015). doi:10.1007/978-3-319-19890-3_28
7. Thalhammer, A., Rettinger, A.: ELES: combining entity linking and entity summarization. In: Bozzon, A., Cudre-Maroux, P., Pautasso, C. (eds.) ICWE 2016. LNCS, vol. 9671, pp. 547–550. Springer, Heidelberg (2016). doi:10.1007/978-3-319-38791-8_45

8. Thalhammer, A., Lasierra, N., Rettinger, A.: LinkSUM: using link analysis to summarize entity data. In: Bozzon, A., Cudre-Maroux, P., Pautasso, C. (eds.) ICWE 2016. LNCS, vol. 9671, pp. 244–261. Springer, Heidelberg (2016). doi:10.1007/978-3-319-38791-8_14
9. Sydow, M., Pikuta, M., Schenkel, R.: Entity summarisation with limited edge budget on knowledge graphs. In: Computer Science and Information Technology (IMCSIT), pp. 513–516 (2010)
10. Sydow, M., Pikuta, M., Schenkel, R.: The notion of diversity in graphical entity summarisation on semantic knowledge graphs. J. Intell. Inf. Syst. **41**(2), 109–149 (2013)
11. Cheng, G., Tran, T., Qu, Y.: RELIN: relatedness and informativeness-based centrality for entity summarization. In: The Semantic Web - ISWC 2011: 10th International Semantic Web Conference, Bonn, Germany (2011)
12. Waitelonis, J., Sack, H.: Towards exploratory video search using linked data. Multimed. Tools Appl. **59**(2), 645–672 (2012)
13. Thalhammer, A., Toma, I., RoaValverde, A.J., Fensel, D.: Leveraging usage data for linked data movie entity summarization. In: The 21st International World Wide Web Conference, Lyon, France (2012)
14. Singhal, A.: Introducing the Knowledge Graph: things, not strings (2012). https://googleblog.blogspot.com/2012/05/introducing-knowledge-graph-things-not.html
15. Torzec, N.: Yahoo's Knowledge Graph (2014). http://semtechbizsj2014.semanticweb.com/sessionPop.cfm?confid=82&proposalid=6452
16. Qian, R.: Understand your world with bing (2013). http://blogs.bing.com/search/2013/03/21/understand-your-world-with-bing/
17. Rey-Lpez, M., Daz-Redondo, R.P., Fernndez-Vilas, A. et al.: T-MAESTRO and its authoring tool: using adaptation to integrate entertainment into personalized t-learning. Multimed. Tools Appl. **40**, 409–451 (2008)
18. Sydow, M., Pikula, M., Schenkel, R.: DIVERSUM: Towards diversified summarisation of entities in knowledge graphs. In: Data Engineering Workshops (ICDEW), pp. 221–226 (2010)

A Hybrid Embedded-Filter Method for Improving Feature Selection Stability of Random Forests

Wassila Jerbi[1(✉)], Afef Ben Brahim[2(✉)], and Nadia Essoussi[1(✉)]

[1] LARODEC, Institut Supérieur de Gestion, Université de Tunis, Avenue de la Liberté, 2000 Le Bardo, Tunisie
jerbi_wassila@hotmail.fr, nadia.essoussi@isg.rnu.tn
[2] LARODEC, Tunis Business School, Université de Tunis, El Mourouj 2074, Tunisie
afef.benbrahim@yahoo.fr

Abstract. Many domains deal with high dimensional data that are described with few observations compared to the large number of features. Feature selection is frequently used as a pre-processing step to make mining such data more efficient. Actually, the issue of feature selection concerns the stability which consists on the study of the sensibility of selected features to variations in the training set. Random forests are one of the classification algorithms that are also considered as embedded feature selection methods thanks to the selection that occurs in the learning algorithm. However, this method suffers from instability of selection. The purpose of our work is to investigate the classification and feature selection properties of Random Forests. We will have a particular focus on enhancing stability of this algorithm as an embedded feature selection method. A hybrid filter-embedded version of this algorithm is proposed and results show its efficiency.

Keywords: Stability · Feature selection · Classification · High dimensional data · Random forests

1 Introduction

With the evolution of technology that keeps skyrocketing, we are devastated by tremendous amount of data. Therefore, new challenges are imposed in the machine learning field, which have to deal with a large amount of data in their different forms. Besides, recently we are talking more and more about two phenomenons, big data and high dimensional data. Which have drawn several researches. In fact, the increased volume of those type of data makes the learning process painful and less efficient. Henceforth, the pre-processing procedure is crucial to get more adapted data for learning algorithms. Evidently, the more numerous are the features, the higher is the risk of noisiness. For that cause, feature selection is an important step to pre-process data.

A. Abraham et al. (eds.), *Proceedings of the 16th International Conference on Hybrid Intelligent Systems (HIS 2016)*, Advances in Intelligent Systems and Computing 552,
DOI 10.1007/978-3-319-52941-7_37

Feature selection consists on minimizing as possible the number of features while keeping the original properties of data. It is the process of removing redundant and irrelevant features to make the learning algorithm more efficient, so that we can get better results.

Various researches have been interested on feature selection. However, these researches have focused on enhancing the predictive performance while neglecting the stability issue.

Stability is about preserving at most the same result concerning the selected feature subset, avoiding the variation in the selected set of features, even with small changes in the Data set.

In the present work, we look to get a robust feature selection. For that aim, we will investigate one of the ensemble feature selection techniques, namely Random Forest (RF) [4]. Indeed, we will provide some insights about RF and how it behaves as a feature selection technique. We will underline the instability problem which is caused by the intrinsic randomness in the algorithm design.

We will propose a solution to deal with that issue to get more robust feature selection.

2 Dimensionnality Reduction by Feature Selection

Interested in real world data, information industry collects huge data that are often complex and impractical for use. In several domains experts are faced to high dimensional data sets which have huge number of features and few number of observations. Therefore, mining in such forms of data is becoming a challenging task.

For better mining results we have to improve data quality by pre-processing data [8]. Feature selection is an important step in this process. Feature selection reduces the number of features under consideration by removing redundant, irrelevant or weakly relevant features which do not contribute on the classification process [8].

The crucial challenge with feature selection is about how to retain the minimum number of parameters which present the pertinent properties of the data [13], and how to preserve the original meaning of the data making interpretation more feasible. Besides, how can we uncover unlike features? The attribute selection techniques are various. They essentially divide into wrappers, filters, and embedded.

2.1 Filters

Filters assign a score for each attribute to obtain a feature ranking and then select the best subset of features. As they do not depend on a specific type of predictive model, they only take into consideration the intrinsic characteristics of the data [7].

Known to be not time consuming, they are faster than wrappers, but compared to embedded methods they have shown to be competitive in that

respect [7]. A commonly used filter method is the t-test. Which is a filter technique that assigns a score for each attribute. It is generally used to compare two normally distributed samples of population. The t-test is a statistical method that works better with features which have a maximal difference of mean value between groups and a minimal variability within each group [7].

2.2 Wrappers

Wrapper methods consist on assigning a score relative to the usefulness of the subsets of features. Wrappers consider a learning machine algorithm as a black box, it works as follow: firstly, a search algorithm gives a set of features that will be evaluated later by measuring the performance given by the learning algorithm. Thereafter, the set of selected features will be returned back for a next search iteration if it does not reach the required quality. However, if the classifier gives a good predictive performance the subset of features will be returned as the selected set of features. Iterative models like wrappers require massive amounts of computation. They are criticized to be time consuming. In addition, they suffer from the lack of generality since the resulted set of features depends on a specific classifier [7].

2.3 Embedded

As for wrappers, embedded methods depend on a specific learning algorithm. Besides, while the search and evaluation procedures are separated in the wrappers, the embedded method performs feature selection into the classifier construction using its internal parameters. Therefore, they are faster than wrappers and they are more efficient as they avoid the use of all the available data by not needing to divide it into a training set and a validation set [7]. Decision trees such as CART are famous example of embedded methods.

Decision tree induction for feature selection: Decision tree algorithms are often applied for classification task but they are also used as an embedded feature selection method. At each step, an attribute is chosen following an evaluation measure, in order to be represented in one of the tree nodes. Iteratively, the best feature is selected according to its discriminative power of separating different classes. This procedure is repeated until reaching some stopping criterion. Consequently, the obtained model is a tree that uses a specific subset of features. To put it differently, the reduced set of features can be determined from features appearing in nodes. Thus, feature selection has been performed implicitly into the algorithm [8]. Perceived as an ensemble form of decision trees, thus RF are also considered as an embedded feature selection.

3 Random Forests for Feature Selection

Aiming to get better results, the principle of ensemble learning has been proposed to enhance the prediction performance in the machine learning field. Ensemble

learning [5] relies on constructing a set of classifiers dealing with the same problem. Thus, to classify a new object, predictions made by individual classifiers, are combined by voting or averaging to get a final consensus [5].

In that context, looking to enhance classification performance, RF have been proposed by Breiman [4]. In fact, RF revolves around the generic principle of classifiers combination. It consists basically on generating a large number of trees, to let them later vote for the most popular class. In order to grow these ensembles, iteratively, we take a bootstrap sample from the available Data set to govern the construction of each tree in the ensemble. Then, to determine the split at each node, a random selection of features is applied [4].

RF gained a great interest as an ensemble method in the machine learning field [4], because they have significantly improved the classification performance and have remarkably proved its efficiency compared to single tree classifier [1]. For that reason the idea has been extended and has been adopted in the feature selection domain.

In addition, we can easily remark that the RF algorithm proceeds by selecting features which improve the most the predictive performance to put them in the tree nodes [4]. Under those circumstances, RF is considered as an embedded feature selection method which yields to high prediction accuracy.

3.1 Stability Issue of Random Forests

In several domains databases are constantly updated. Hence, many changes occur regularly whether with adding more observations or new features. Consequently, those small adjustments in the data set cause remarkable variation in the selected set of features and that is not practical for use. This occurs specially with data with small instances compared to the number of features [9]. Stability of feature selection is defined as the sensitivity of a given process to variations in the training set [9] i.e., it is obtained by measuring the similarity between different resulted set of features. In fact, it is about keeping at most the same resulted feature subset, even with small changes in the Data set. Many researches have been interested on stability of feature selection for high dimensional data, using the ensemble learning concept [11] or hybrid methods [3].

As described before, RF introduced by Leo Breiman [4] is an ensemble learning method that intends to enhance the predictive performance.

Here, we talk about its performance as an embedded feature selection due to selection that already occurs at each node of the tree. To be considered as an efficient feature selection method, it must satisfy the two criteria of a robust feature selection method, which are the classification performance and the stability.

In addition to the small sample size, RF use random components to generate diversity in the ensemble of trees and this is also a source of instability. Applied experiments on high dimensional data [10], have proved that the selected best features with RF changes dramatically even with little change on data. Therefore, used as a feature selection method RF have shown good results in prediction accuracy, whereas still the problem of stability which have not been resolved.

4 Hybrid Embedded-Filter Feature Selection

To speak about robust feature selection we have to assess the prediction performance and the stability of the selected features. So it is about finding a consensus between the stability of feature selection and the prediction accuracy, as they are both important for classification task. For that goal, we choose an algorithm that have already shown its performance in terms of prediction accuracy, and we will try to improve its stability index. In fact, the idea of our proposal is to take advantage of the standard RF algorithm to ensure a good classification accuracy, then working on improving its stability on selecting features, while preserving at most the high prediction accuracy.

The instability of RF is caused by random components involved in the learning algorithm. At first, the bagging method is used to select a bootstrap sample for each tree growth then the randomization is used to choose features for each split. Those two steps developed in the algorithm make harder to reproduce the same feature selection. However, the randomization is required to achieve diversity of trees in order to assure later, efficient results. Still, as we work on high dimensional data, even a little variation causes a significant change in the selection of a feature. Thus, bagging is very sufficient to get diverse trees. Hence, in order to improve the feature selection stability of RF we have to focus on the randomization and try to work on reducing its effects.

In our proposed method, we make the preference to eliminate one of the two random components and act through the choice of the split node. To make it clearer, in the basic RF algorithm, each time to find the split node, a number of features are selected randomly. In our approach, as argumented we look to

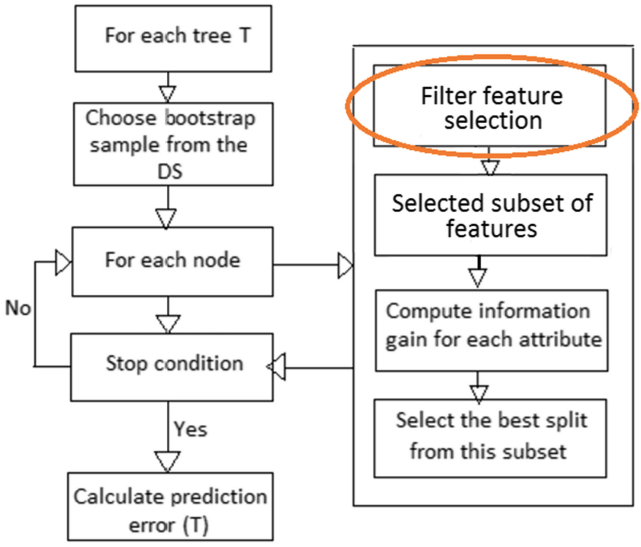


Fig. 1. Flowchart: hybrid random forests.

eliminate randomness at this level, so we proceed by replacing the random choice of the subset of features by introducing a filter method. Consequently, these features will be selected using a filter method instead of randomness. Figure 1 describes the process of these hybrid approach. Notably, we look to minimize the randomness to get more stable selection.

Furthermore, we will also propose different combination methods trying to more evaluate the stability. Accordingly, at first, features are ranked according to their importance on the classification process and we preserve only 1% of the top ranked features to test the stability. Then, we will proceed by diversifying the number of features with different combination strategies; top ranked features and weight aggregation.

5 Experimental Study

As described above, our proposal consists on replacing the random selection which occurs for the split node by a filter selection method. For this purpose, in our hybrid feature selection experiments we introduce the t-test, one of the filter selection methods known to be fast and efficient [7]. To better analyze the performance of our proposal, we evaluate the resulted hybrid RF model for different settings where we variate the number of trees in the forest (nTrees) from 5 to 500.

To assess our approach we use 10 fold cross validation consisting on using 90% of the data as a training set. This protocol allows us simulating small changes in the data, so we get diverse training sets to generate different RF. Then we perform a comparison between results obtained by the basic RF algorithm and the proposed hybrid RF algorithm.

In a second experimental setting, to evaluate our approach we apply two different methods that will serve to combine the feature subsets obtained by each forest. For that, we use the top rank method and the rank aggregation method. Our aim is to see whether the combination technique affects results with the focus on obtaining the best stability.

5.1 Data Sets

Three data sets are used in this experimental process; Lymphoma, Bladder and DLBCL. Characterized by thousands of features and tens of instances, we employ this specific type of data sets known as 'high dimensional data' because they present great challenges for classification and feature selection algorithms. Table 1 contains brief description of the used databases.

5.2 Evaluation Metrics

To evaluate our approach we use the stability measure (Kuncheva). This index ranges within [−1,1], the higher is this index the more are common features across the different sets. As we can not work on improving the stability measure while neglecting the prediction accuracy, we adopt the (F-measure) which is a performance evaluator that describes the harmonic mean of precision and recall.

Table 1. Data set characteristics.

Name	#instances	#Features	References
Lymphoma	45	4026	[2]
Bladder	31	3036	[6]
DLBCL	77	7029	[12]

5.3 Results

To estimate the robustness of the embedded-filter feature selection we proceed as explained above. We run both algorithms (basic RF and hybrid RF) across different data sets. Only 1% of features are chosen to test stability i.e., for lymphoma data set we keep 40 features from the top ranked features, for bladder data base the number of retained features will be 30 and for DLBCL we perserve 70 features.

Table 2 shows the stability and prediction results while running the basic RF algorithm and then, the hybrid algorithm (RF with t-test). We can easily see the increase of stability for all data sets with the hybrid version using the t-test. However, we remark that F-measure is slightly better when running the basic RF algorithm. At first sight, some can argue that the increase in stability, comes at cost of lower accuracies. But once focusing more in the results we can deduce that the tiny decrease of F-measure can be neglected via the significant increase of stability. So, we can deduce that the predictive performance is at most the same while the stability has remarkably improved.

It can be observed that the hybrid feature selection algorithm (RF with t-test) provides more robust feature selection comparing to the basic algorithm (RF). However, the difference on the stability is dependent of the data set and other parameters like the number of trees in the forest (nTrees). Besides, it is

Table 2. 10 fold CV F-measure and Kuncheva index for stability measure are evaluated with data variation.

Data set			nTrees=5	nTrees=50	nTrees=100	nTrees=200	nTrees=500
Lymph	RF (basic Algo)	Fm	0.7826	0.9111	0.9583	0.9362	0.9565
		Stab	0.6482	0.1566	0.1746	0.1723	0.1712
	RF(with t-test)	Fm	0.7755	0.9167	0.9362	0.9131	0.9235
		Stab	0.8367	0.4041	0.5264	0.4810	0.4822
Bladder	RF (basic Algo)	Fm	0.8182	0.8696	0.8696	0.8696	0.9091
		Stab	0.7598	0.1187	0.1269	0.1157	0.1441
	RF(with t-test)	Fm	0.8000	0.8571	0.8571	0.8571	0.8901
		Stab	0.8526	0.3379	0.3237	0.3663	0.3973
DLBCL	RF (basic Algo)	Fm	0.9106	0.9106	0.9421	0.9180	0.9470
		Stab	0.7246	0.0958	0.1610	0.1916	0.1998
	RF(with t-test)	Fm	0.9500	0.9076	0.9412	0.9178	0.9468
		Stab	0.7486	0.4071	0.4521	0.4989	0.5049

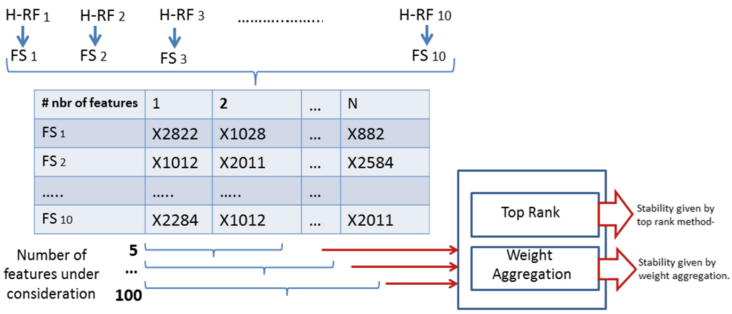


Fig. 2. Testing different feature combination techniques while variating the number of features under consideration.

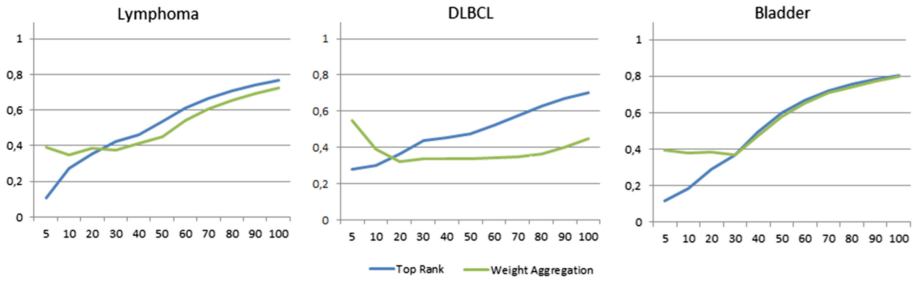


Fig. 3. Stability results of Hybrid RF using the top rank and weight aggregation methods with various feature cardinality.

important to underline that for all data sets the stability measure increases while running the new approach but with various expansion rates.

Aiming to more evaluate the stability of our proposal, we investigate other strategy. As described in Fig. 2, The weight aggregation and the top rank methods are tested with various number of features.

Looking to get a higher Kuncheva index. The results given by applying these proceeds at the three data sets are illustrated in Fig. 3. Let's denote that FS_n refers to the Feature Selection obtained from the hybrid RF $H-RF_n$. In the first part of the experiments, to evaluate stability we proceed by considering only 1% of the top ranked features while diversifying the ensemble size. In this part, as described in Fig. 2, we have fixed the ensemble size at 50 trees, and we tested the stability results with various feature cardinality [5...100] using top ranked and weight aggregating methods.

As it can be observed Fig. 3, the weight aggregation leads to better results than top rank methods with a number of features lower than 20. However, when the number of features under consideration increases, the stability with top rank process exceeds the one obtained by weight aggregation. Moreover, the stability of our hybrid random forests measured with kuncheva index is improved while

increasing the number of retained features, and reaches 0.8 using 100 features for Bladder data set (Fig. 3).

So that, we can conclude that our embedded-filter feature selection provides high stability measure while considering a substantial number of features. Furthermore, it is important to underline that in addition to these high stability results we have obtained a good classification performance that varies between 0.8 and 0.9.

Under those circumstances, we get through hybrid method to deliver a modified random forests that provide a stable feature selection while maintaining a good classification performance.

6 Conclusion

In this work, our objective was to obtain a robust feature selection. We have proposed an hybrid approach, based on joining both embedded and filter feature selection methods. The main idea of our proposal is to take advantage of the standard random forests algorithm to ensure a good classification accuracy, then working on improving its stability on selecting features. For this purpose, we used to reduce the effect of randomization involved in the standard algorithm. So, instead of proceeding by choosing randomly a number of features we proceed by applying a filter feature selection (t-test). Our experimental study is a combined analyze of predictive performance and robustness of feature selection. The experimentation shows that our hybrid method, based on joining filter method (t-test) to embedded method (RF), finds a trade-off between these two important criteria. Thus, our initial objective is satisfied proving the efficiency of our proposal.

References

1. Ali, J., Khan, R., Ahmad, N., Maqsood, I.: Random forests and decision trees. Int. J. Comput. Sci. Issues (IJCSI) **9**(5), 1–7 (2012)
2. Alizadeh, A.A., Eisen, M.B., Davis, R.E., Ma, C., Lossos, I.S., Rosenwald, A., Boldrick, J.C., Sabet, H., Tran, T., Yu, X., et al.: Distinct types of diffuse large B-cell lymphoma identified by gene expression profiling. Nature **403**(6769), 503–511 (2000)
3. Ben Brahim, A., Limam, M.: A hybrid feature selection method based on instance learning and cooperative subset search. Pattern Recogn. Lett. **69**(C), 28–34 (2016)
4. Breiman, L.: Random forests. Mach. Learn. **45**(1), 5–32 (2001)
5. Dietterich, T.G.: Ensemble methods in machine learning. In: Kittler, J., Roli, F. (eds.) MCS 2000. LNCS, vol. 1857, pp. 1–15. Springer, Heidelberg (2000). doi:10.1007/3-540-45014-9_1
6. Dyrskjøt, L., Thykjaer, T., Kruhøffer, M., Jensen, J.L., Marcussen, N., Hamilton-Dutoit, S., Wolf, H., Ørntoft, T.F.: Identifying distinct classes of bladder carcinoma using microarrays. Nat. Genet. **33**(1), 90–96 (2003)
7. Guyon, I., Elisseeff, A.: An introduction to variable and feature selection. J. Mach. Learn. Res. **3**, 1157–1182 (2003)

8. Han, J., Kamber, M.: Data Mining: Concepts and Techniques. Data Management Systems. Morgan Kaufmann, San Francisco (2000)
9. Kalousis, A., Prados, J., Hilario, M.: Stability of feature selection algorithms: a study on high-dimensional spaces. Knowl. Inf. Syst. **12**(1), 95–116 (2007)
10. Li, S., Harner, E.J., Adjeroh, D.A.: Random KNN feature selection-a fast and stable alternative to random forests. BMC Bioinformatics **12**(1), 1 (2011)
11. Saeys, Y., Abeel, T., Van de Peer, Y.: Robust feature selection using ensemble feature selection techniques. In: Daelemans, W., Goethals, B., Morik, K. (eds.) ECML PKDD 2008. LNCS (LNAI), vol. 5212, pp. 313–325. Springer, Heidelberg (2008). doi:10.1007/978-3-540-87481-2_21
12. Shipp, M.A., Ross, K.N., Tamayo, P., Weng, A.P., Kutok, J.L., Aguiar, R.C., Gaasenbeek, M., Angelo, M., Reich, M., Pinkus, G.S., et al.: Diffuse large B-cell lymphoma outcome prediction by gene-expression profiling and supervised machine learning. Nat. Med. **8**(1), 68–74 (2002)
13. van der Maaten, L.J.P., van den Herik, H.J.: Dimensionality reduction: A comparative review. Technical report. Tilburg Centre for Creative Computing, Tilburg University, Tilburg, Netherlands Technical Report: 2009–005 (2009)

Grey Wolf Optimizer for Training Elman Neural Network

Besma Rabhi[1(✉)], Habib Dhahri[2], Adel M. Alimi[3], and Fahd A. Alturki[4]

[1] Faculty of Sciences and Techniques of Sidi Bouzid, University of Kairouan, Kairouan, Tunisia
besma.rebhi.2015@ieee.org
[2] College of Computer and Information Sciences, King Saud University, Riyadh, Saudi Arabia
[3] National School of Engineers of Sfax, University of Sfax, Sfax, Tunisia
[4] Faculty of Engineering, King Saud University, Riyadh, Saudi Arabia

Abstract. In this paper, we apply the Elman Neural Network (ENN) trained with Grey Wolf Optimizer (GWO) for time series predictions and data classification. The Grey Wolf Optimizer algorithm optimizes the network parameters. In order to evaluate the performance of the proposed method, we have carried out some experiments on two data sets: Mackey Glass, and Breast Cancer. We also give simulation examples to compare the effectiveness of the model with five known meta-heuristics methods in the literature. The results show that the GWO-ENN model produces a better generalization performance.

Keywords: ENN · Grey wolf optimizer · Data classification · Time series predictions

1 Introduction

In recent years, Neural Network (NN) has been extensively and successfully used as a tool in many practical problems, especially in pattern recognition [19], time series prediction [20] and approximation function [21]. Recurrent Neural Network (RNN) is a type of NN [1] characterized by the feedback of information. Therefore, RNN is common in learning, so, trained a NN to be able to learn from different examples. In actual fact, there are two types of learning: Supervised [2] and unsupervised [3]. In the first case, learning is based on an external source, whereas, in the second type, the NN learn without any external information. In supervised learning, training neural network is an important task to get the best performance to knowing new examples that are different to those used as inputs. Optimization can be performed by nature-inspired optimization methods which have become an efficient choice compared to other optimization methods [4]. Some of the most popular method are: Ant Colony Optimization (ACO) [7], Genetic Algorithm (GA) [5], Particle Swarm Optimization (PSO) [6], Evolutionary Strategies (ES) [8] and Population Based Incremental Learning (PBIL) [9]. The strength of these algorithms is their ability to approximate to the global optimum.

A. Abraham et al. (eds.), *Proceedings of the 16th International Conference on Hybrid Intelligent Systems (HIS 2016)*, Advances in Intelligent Systems and Computing 552,
DOI 10.1007/978-3-319-52941-7_38

Grey Wolf Optimizer (GWO) is a recent technique based on population swarm intelligence, inspired by the nature of the gray wolf. In [10], the author demonstrates the superiority, effectiveness and performance of Modified GWO compared with other existing heuristics methods (PSO, GA, ACO, ES) treating the problem of wind speed forecasting and use Elman NN as RNN architecture. [11] shows that the improved GWO is very appropriate to be used in **q**Gaussian Radial Basis Functional-link nets (**q**RBFLNs) neural networks, treating the multiclass classification problem. In this paper, we use GWO to find the most optimized parameters of ENN.

We also verify the exploration and exploitation of this algorithm on training ENN for the classification and prediction tasks compared with other heuristics method.

The rest of the paper is detailed as follows: In Sect. 2 we present the Elman NN (ENN); In the Sect. 3, we explain the main function of GWO trained ENN, The experimental results will be given in the Sect. 4 and finally in the Sect. 5, we give the conclusion.

2 Elman Neural Network

Elman Neural Network (ENN) proposed in [12], is the most widely used RNN architecture. Its structure is chosen over the Jordan [13] network thanks to its hidden layer being wider than the output layer. This wider layer allows more values to be fed back to the input, consequently authorizing more information to be available to the network [14]. Its design is: the input layer, the hidden layer, the recurrent link known as context layer and the output layer.

x_i (i = 1…m) is the input vector, y_k is the output of ENN and z_j (j = 1…n) is the output of hidden layer. b_j and b_k are the biases in the hidden layer and the output layer respectively. u_j denotes the context layer neurons. w_{ij} is the weight that connects between the input nodes (i) and the hidden nodes (j). c_j denotes the weight that connects between the hidden nodes and the context nodes. v_{jk} is the weight that connects the node j in the hidden layer to the output nodes.

$$net_j(t) = \sum_{i=1}^{m} w_{ij} x_i(t-1) + \sum_{j=1}^{n} c_j u_j(t) + b_j. \quad (1)$$

u_j is the context node value, calculated by (2)

$$u_j(t) = z_j(t-1). \quad (2)$$

The activation function selected in hidden layer is the sigmoid function, which is given as follows:

$$z_j(t) = f\ (net_j(t)). \quad (3)$$

$$f\ (x) = 1/(1 + e^{-x})$$

The output of ENN is given as follows:

$$\begin{aligned} net_k(t) &= \sum_{j=1}^{n} v_{jk} z_j(t) + b_k. \\ y_k(t) &= f\ (net_k(t)) \end{aligned} \tag{4}$$

The architecture of ENN is presented in Fig. 1.

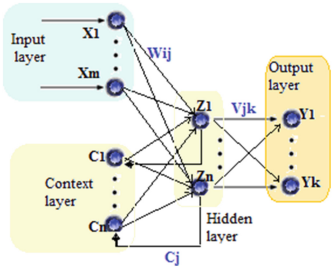


Fig. 1. Elman Neural Network architecture.

3 GWO Trained Elman NN

The GWO algorithm was proposed by [15]. As it's a meta-heuristic algorithm, the basic idea of GWO was inspired by grey wolves in nature. Recently, GWO algorithm has improved its success and performance supply global optimal solution in different problems such as in [10, 11]. The general idea of this algorithm is to devised population into four groups: alpha (α), beta (β), delta (δ) and omega (ω). The promise area of the search space is supplied by the wolves (ω) guided by the first three fittest wolves (α, β, δ). For each optimization step, the wolves relocate their position around α, β and δ as below:

$$D = |G \cdot Xp(i) - Xw(i)| \tag{5}$$

$$Xw(i+1) = Xp(i) - H \cdot D \tag{6}$$

Where i denotes the current iteration, Xp is the current position vector of the prey, Xw is the position of the grey wolf. G and H are coefficient vectors which are enumerated as: $G = 2 \cdot r_2$ and $H = 2u \cdot r_1 u$.

u is linearly decreased from 2 to 0, r_1 and r_2 denotes random vector in [0,1]. The main role of these coefficients is to help the wolves to update their position around the prey. It is constantly given that α, β and δ are the most closer to the prey. This requires the allocation of the first three best positions obtained during optimization as α, β and δ respectively. The rest of wolves (ω) relocate their position with respect to α, β and δ which are presented as follows:

$$D\alpha = |G1 \cdot X\alpha - Xw| \tag{7}$$

$$D\beta = |G2 \cdot X\beta - Xw| \tag{8}$$

$$D\delta = |G3 \cdot X\delta - Xw| \tag{9}$$

$$Xw1 = X\alpha - H1 \cdot D\alpha \tag{10}$$

$$Xw2 = X\beta - H2 \cdot D\beta \tag{11}$$

$$Xw3 = X\delta - H3 \cdot D\delta \tag{12}$$

$$Xw(i+1) = \frac{Xw1 + Xw2 + Xw3}{3} \tag{13}$$

Where, Xα denotes the first best position, Xβ is the second best position and Xδ shows the third best position. G1, G2 and G3 are random coefficients. After calculating the distance between the current position and α, β and δ respectively as in Eqs. (7), (8) and (9) it's necessary to estimate the new position of the present wolves as in Eqs. (10) to (13).

Where, Xα, Xβ and Xδ presents the position of α, β and δ respectively. H1, H2, H3 are random coefficients. Generally, GWO algorithm can be defined by the following steps [15]:

1- Initialize population values of wolves randomly.
2- Calculate the fitness values for each wolf.
3- Save the first three best solutions as α, β and δ respectively.
4- Relocate the position of the rest of wolves (ω) based on Eqs. (5) to (13).
5- Update parameters value (u, G, H).
6- If not satisfying criterion, go to the second step.
7- Store the position of alpha as global optimum solution.

The GWO algorithm is used to an ENN to finding best combination of biases and weights capitalized on two phases:

- Representing weights and biases.
- Computing objective value defined by the error of ENN to evaluate position wolves performance.

3.1 Representing Scheme of ENN Trained by GWO

Encoding scheme is the most important step in training an ENN by meta-heuristics algorithms. Thus in GWO, the variables (weights and biases) are presented in the form of a vector. As an example of this structure, the final scheme of ENN presented in Fig. 2 is as below:

ENN = {W, O} = {W13, W23, W43, W35, b1, b2}

This figure shows that there are two nodes for the input layer, one node for the hidden and the output layer. W13 and W23 denote the weights between input nodes

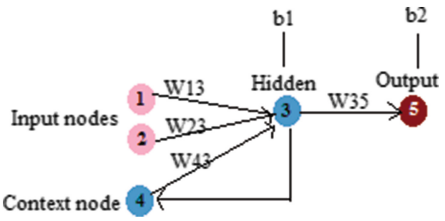


Fig. 2. ENN with 2 inputs, 1 hidden and 1 output.

and hidden node. W43 presents the weight between context and hidden node. W35 shows the weight between hidden and output node. b1 and b2 are the biases of hidden and output node respectively.

3.2 Objective Function

After initializing the variables (weight and biases), we must define the fitness function of the GWO algorithm. In this work the Mean Square Error (MSE) is used to compute the output error as objective function.

$$\mathrm{MSE(ENN)} = \sum_{K=1}^{S} \frac{\sum_{K=1}^{m} \left(o_i^k - d_i^k\right)^2}{S}. \tag{14}$$

Where S is the number of training samples, m denotes the number of output, o_i^k is the obtained output of the *i*th input unit when the *k*th training sample seems in the input and d denotes the desired output.

After all, the main goal of training ENN is to minimize the MSE values for all training samples. The GWO algorithm assumes the weights and biases as populations and modifies them to optimize the error rate and to achieve the best performance in training and testing steps.

In fact, the first step of the proposed model is to initialize randomly weights and biases as population or wolves. The second step is to calculate MSE for each wolf by Eq. (14) to distinguish between the best and the worst parameters, after then, we must mark the first three good solutions as α, β and δ respectively. The next step is to modify the position of the rest of wolves (ω) based on Eqs. (5) to (13). These steps are repeated until satisfaction of the stop condition which can be the number of iterations or the error rate and then, store the position of alpha as global optimum solution. To more explicate the main functionality of GWO-ENN, Fig. 3 presents a conceptual picture.

As may be seen in this figure, there are four groups (based on their fitness values): alpha (α), beta (β), delta (δ) and omega (ω). α, β and δ present the first three good solutions. Whereas the group of ω wolves provides a highest MSE, it presents the poor solution. So it's more likely to relocate it toward α, β and δ groups. In fact, α wolf is authorized more than others to interfere to modify the weights and biases of ω wolves that's why the blue color (α) is more dominated then red (β) and green (δ) in the new ω wolf position. Obviously, the weights and biases values are not totally modified by α, β

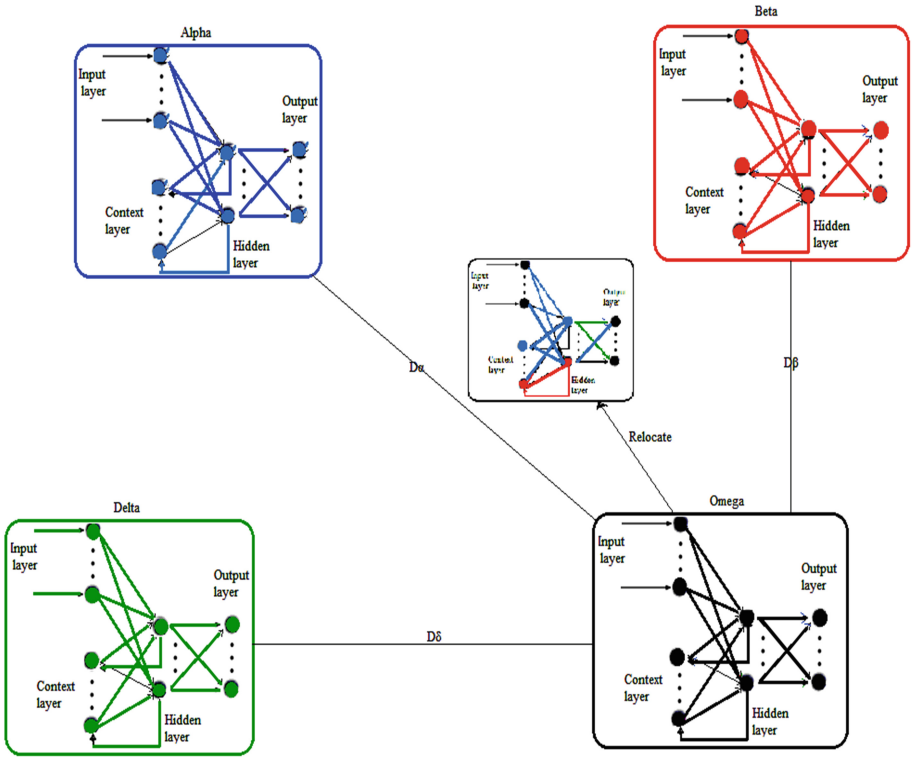


Fig. 3. Conceptual picture of GWO-ENN model.

and δ, so a value is calculated based on Eq. (13). The highlight of GWO algorithm is to have better solutions than the first because the weight and bias changed toward the good ENN values in each iteration.

4 Result Analysis and Discussion

In this section, the proposed GWO-based Elman NN is verified on two benchmark problems: Breast Cancer [16], for data classification and Mackey Glass [17] for time series prediction problem. The classification data sets based on two performance criteria: (a) MSE value and (b) classification accuracy. The time series prediction dataset is based also on two performance criteria: MSE error for training and testing phase. In this work, we are not forced to find the best parameters. In fact, the increase in population size and the number of iteration, would improve the test error, but in this work we are interested in comparing the performance in terms of avoiding local minima and what is the algorithm that converges faster towards the global minima. Thus, just use the same network parameters such as number of nodes, the value of weight initialization and size of population. The transform function used is log-sigmoid

activation function. For all algorithms we initialize the optimized parameters randomly in the range [–10, 10]. We simulate these algorithms 30 times, for 200 generations.

According to [18] there is no standard rule for determining the suitable number of hidden nodes. We fixed it on the basis of this theorem "One hidden layer and 2N + 1 hidden neurons Sufficient for N inputs". Table 1 shows the different number of input, hidden and output node of each datasets.

Table 1. Structure of each datasets

Datasets	Number of input nodes	Number of hidden nodes	Number of output nodes
Breast Cancer	9	19	1
Mackey Glass	4	9	1

The initial parameters of meta-heuristics algorithms are fixed in Table 2; it shows various initialization settings for the optimization methods.

Table 2. Parameters settings of algorithms

Method	Parameter	Value
GWO	U	From 2 to 0
ACO	Initial pheromone (τ0)	1e–06
	Pheromone update constant (Q)	20
GA	Selection mechanism	Roulette wheel
	Crossover probability	1
PSO	Cognitive constant (c1)	1
	Inertia weight (w)	0.3
ES	Λ	10
	$\sum$	1
PBIL	Learning rate	0.05

4.1 Breast Cancer

This dataset was obtained from the University of Winsconsin Hospitals, Madison, established by William H. Wolberg. The output convergence of GWO algorithm is presented in Fig. 4.

Table 3 presents the results of training algorithms to solve this problem. It illustrates the comparison between performances of GWO-ENN with GA, PSO, ACO, ES and PBIL algorithms. From Table 3, it can be easier to show that the proposed algorithm achieves less MSE (0.0012175) and high accuracy (99.99).

MSE value indicates that GWO Elman NN has the best ability to converge to global optimum among the other methods which demonstrates the efficacy of GWO-ENN for data classification.

From these results, it can be easier to understand that the proposed algorithm achieves higher performances than the other methods in terms of MSE and accuracy.

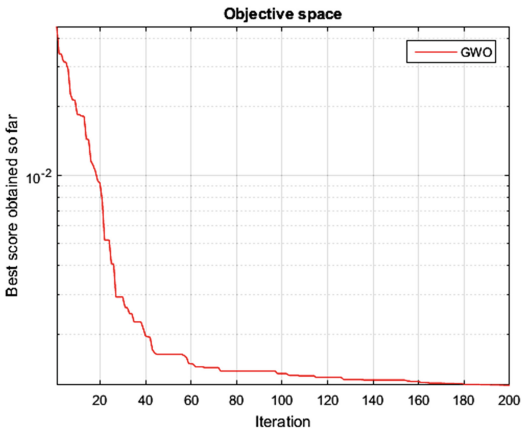


Fig. 4. Convergence Breast Cancer Elman NN.

Table 3. Experimental results for Breast Cancer dataset

Algorithms	MSE error	Accuracy
GWO	0. 0011948	99.99
GA	0.0025149	98.45
PSO	0.0043705	94.50
ACO	0.0073633	76.25
ES	0.0062843	73.00
PBIL	0.0320012	03.99

4.2 Mackey–Glass Time Series Prediction

An application of GWO-ENN to the Mackey-Glass time series prediction is done, using the following equation:

$$\frac{dx(t)}{dt} = \frac{ax(t-\tau)}{1+x^c(t-\tau)} - bx(t) \tag{15}$$

In our work, the input of ENN is four data points: x(t), x(t – 6), x(t – 12) and x (t – 18). The output is presented in Eq. 16:

$$x(t+6) = f(x(t), x(t-6), x(t-12), x(t-18)) \tag{16}$$

After 200 iterations of the training process, the convergence curve of GWO algorithm is summarized in Fig. 5. Table 4 illustrates the MSE error for training phase and the other for testing phase of each algorithm. From this table it's clear that our method achieves low MSE value for both training and testing phase (0.00712, 0.007102). However, the other algorithms, as, PSO, GA, ACO, ES and PBIL have a MSE values quite larger than our model. Similarly, Fig. 5 represents the MSE

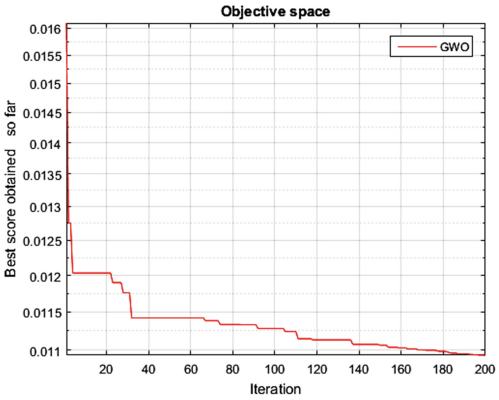


Fig. 5. Convergence Mackey-Glass Elman NN.

Table 4. Experimental results for Mackey-Glass dataset

Algorithms	MSE error training	MSE error test
GWO	0.01019	0.008510
PSO	0.01043	0.071985
GA	0.01130	0.009851
ACO	0.04851	0.093598
ES	0.04748	0.082876
PBIL	0.02293	0.091885

convergence for Mackey-Glass time series prediction. This Figure demonstrates that the proposed GWO-ENN have better result than the other algorithms.

GWO-ENN shows again efficiencies for the prediction of Mackey-Glass time series prediction.

5 Conclusion

In this work, the recently proposed GWO algorithm is proposed to train Elman Neural Network (ENN) for various benchmark problems. The experiment results show that the GWO-ENN model can effectively classify the data such as Breast Cancer data set. Statistical results show that the proposed algorithm outperforms the GA, PSO, EA, ACO and PBIL algorithms. GWO-ENN makes a success in the convergence time and a great performance in avoiding local minima. This algorithm achieves the high level of exploitation and exploration.

Our future work is aimed at optimizing the other structure parameters of ENN such as the number of hidden nodes and applying it in multi-objective optimization ENN.

Acknowledgments. The authors would like to acknowledge the financial support of this work by grants from the General Direction of Scientific Research (DGRST), Tunisia, under the ARUB program01/UR/11/02.

References

1. Dorffner, G.: Neural networks for time series processing. In: Neural Network World (1996)
2. Caruana, R., Niculescu-Mizil, A.: An empirical comparison of supervised learning algorithms. In: Proceedings of the 23rd International Conference on Machine Learning, pp. 161–168 (2006)
3. Wang, D.: Unsupervised learning: foundations of neural computation. AI Mag. **22**(2), 101 (2001)
4. Yoo, D.G., Kim, J.H.: Meta-heuristic algorithms as tools for hydrological science. Geosci. Lett. **1**(1), 1–7 (2014)
5. Pham, D.T., Karaboga, D.: Training Elman and Jordan networks for system identification using genetic algorithms. Artif. Intell. Eng. **13**(2), 107–117 (1999)
6. Xiao, P., Venayagamoorthy, G.K., Corzine, K.A.: Combined training of recurrent neural networks with particle swarm optimization and backpropagation algorithms for impedance identification. In: IEEE Swarm Intelligence Symposium, pp. 9–15 (2007)
7. Zhipeng, Y., Minfang, P., Hao, H., Xianfeng, L.: Fault locating of grounding grids based on ant colony optimizing Elman neural network. In: International Conference on Digital Manufacturing and Automation, pp. 406–409 (2012)
8. Kawada, K., Yamamoto, T., Mada, Y.: A design of evolutionary recurrent neural-net based controllers for an inverted pendulum. In: Control Conference Asian, vol. 5, pp. 1419–1422. IEEE (2004)
9. Palafox, L., Iba, H.: On the use of population based incremental learning to do reverse engineering on gene regulatory networks. In: IEEE Congress on Evolutionary Computation, pp. 1–8 (2012)
10. Madhiarasan, M., Deepa, S.N.: ELMAN neural network with modified grey wolf optimizer for enhanced wind speed forecasting. Circuits Syst. **7**(10), 2975 (2016)
11. Muangkote, N., Sunat, K., Chiewchanwattana, S.: An improved grey wolf optimizer for training q-Gaussian Radial Basis Functional-link nets. In: IEEE Computer Science and Engineering Conference (ICSEC), pp. 209–214 (2014)
12. Elman, J.L.: Finding structure in time. Cogn. Sci. **14**(2), 179–211 (1990)
13. Jordan, M.I.: Serial order: a parallel distributed processing approach. Adv. Psychol. **121**, 471–495 (1997)
14. Welch, R.L., Ruffing, S.M., Venayagamoorthy, G.K.: Comparison of feedforward and feedback neural network architectures for short term wind speed prediction. In: International Joint Conference on Neural Networks, pp. 3335–3340. IEEE (2009)
15. Mirjalili, S., Mirjalili, S.M., Lewis, A.: Grey wolf optimizer. In: Advances in Engineering Software, pp. 46–61 (2014)
16. Wolberg, W.H., Mangasarian, O.L.: Multisurface method of pattern separation for medical diagnosis applied to breast cytology. Proc. Nat. Acad. Sci. **87**(23), 9193–9196 (1990)
17. Mackey, M.C., Glass, L.: Oscillation and chaos in physiological control systems. Science **197**(4300), 287–289 (1977)
18. Shamsuddin, S.M.: Lecture note advanced artificial intelligence: Number of hidden neurons. Unpublished note. Universiti Teknologi Malaysia, Skudai (2004)

19. Alimi, M.A.: The recognition of arabic handwritten characters with the Beta neuro-fuzzy network. In: Proceedings of the 17éme Journées Tunisiennes d'Électrotechnique et d'Automatique, JTEA 1997, Nabeul, Tunisia, vol. 1, pp. 349–356 (1997)
20. Subudhi, B., Jena, D.: A differential evolution based neural network approach to nonlinear system identification. Appl. Soft Comput. **11**(1), 861–871 (2011)
21. Dhahri, H., Alimi, M.A.: The modified differential evolution and the RBF (MDE-RBF) neural network for time series prediction. In: Proceedings of International Joint Conference on Neural Networks IJCNN 2006, Vancouver, pp 5245–5250 (2006)

DNA Sequence Classification Using Power Spectrum and Wavelet Neural Network

Abdesselem Dakhli[1(✉)], Wajdi Bellil[2], and Chokri Ben Amar[2]

[1] Department of Computer Science, REGIM, University of Gabes, 6002 Gabes, Tunisia
abdesselemdakhli@gmail.com
[2] Department of Computer Science, REGIM, University of Sfax, 3018 Sfax, Tunisia
{wajdi.bellil, chokri.benamar}@ieee.org

Abstract. In this paper, we present a new method to cluster DNA sequence. The proposed method is based on using the Power Spectrum and the Wavelet Neural Network (WNN). The satisfying performance of the Wavelet Neural Networks (WNN) depends on an appropriate determination of the WNN structure. Our approach uses the Least Trimmed Square (LTS) to select the wavelet candidates from the Multi Library of the Wavelet Neural Networks (MLWNN) for constructing the WNN. The LTS has been able to optimize the wavelet neural network. The LTS algorithm is to find the regressors, which provide the most significant contribution to the approximation of error reduction. This wavelet can reduce the approximation error.

In this study, the DNA sequence is coded by using a binary format. The Fourier transform is applied to attain respective Power Spectra (PS) by using the binary indicator sequence. The PS is applied to construct the mathematical moments which be used to build the vectors of real numbers, which are applied to compare easily the sequences with different lengths. Our aim is to construct classifier method that gives highly accurate results. This classifier permits to classify the DNA sequence of organisms. The classification results are compared to other classifiers. The experimental results have shown that the WNN-PS model outperformed the other classifier in terms of both the running time and clustering. In this paper, our approach consists of three phases. The first one, which is called transformation, is composed of three sub steps; binary codification of DNA sequences, Fourier Transform and Power Spectrum Signal Processing. The second section is the approximation; it is empowered by the use of Multi Library Wavelet Neural Networks (MLWNN). Finally, the third section, which is called the classification of the DNA sequences. The Euclidean distances is used to classify the signatures of the DNA sequences.

Keywords: WNN · LTS · PS · DNA sequences · MLWNN

1 Introduction

Various approaches are used for clustering the DNA sequences such as the WNN, which is applied to construct a classification system. Cathy H. et al. used an artificial neural network to classify the DNA sequences [10]. Moreover, Agnieska et al. are proposed a

A. Abraham et al. (eds.), *Proceedings of the 16th International Conference on Hybrid Intelligent Systems (HIS 2016)*, Advances in Intelligent Systems and Computing 552,
DOI 10.1007/978-3-319-52941-7_39

method to classify the mitochondrial DNA Sequences. This approach joins the WNN and a Self-Organizing map method. The feature vector sequences constructed by using the WNN [11]. Xiu Wen et al. used a Wavelet packet analysis to extract features of DNA sequences, which are applied to recognize the types of other sequences [12]. C. Wu et al. applied the neural network to classify the nucleic acid sequence. This classifier used three-layer and feed-forward networks that employ back-propagation learning algorithm [13]. Since a DNA sequence can be converted into a sequence of digital signals, the feature vector can be built in time or frequency domains. However, most traditional methods, such as k-tuple and DMK,… models build their feature vectors only in the time domain, i.e., they use direct word sequences [14–19].

The construction of the neural networks structure suffers from some deficiencies: the local minima, the lack of efficient constructive methods, and the convergent efficiency, when using ANNs. As a result, the researchers discovered that the WNN, is a new class of neural networks which joins the wavelet transform approach. The WNN were presented by Benveniste and Zhang. This approach is used to approximate the complex functions with a high rate of convergence [1]. This model has recently attracted extensive attention for its ability to effectively identify nonlinear dynamic systems with incomplete information [1–5]. The satisfying performance of the WNN depends on an appropriate determination of the WNN structure. To solve this task many methods are proposed to optimize the WNN parameters. These methods are applied for training the WNN such as the least-square which is used to train the WNN when outliers are present. These training methods are applied to reduce some function costs and improve performed the approximation quality of the wavelet neural network. On the other hand, the WNN has often been used on a small dimension [6]. The reason is that the complexity of the network structure will exponentially increase with the input dimension. The WNN structure has been studied by several researchers. Moreover, the research effort has been made to deal with this problem over the last decades [6–9]. The application of WNN is usually limited to problem of small dimension. The number of wavelet functions in hidden layer increases with the dimension. Therefore, building and saving WNN of large dimension are of prohibitive cost. Many methods are used to reduce the size of the wavelet neural networks to solve large dimensional task. In this study, we use the Least Trimmed Square (LTS) method to select a little subset of wavelet candidates from MLWNN constructing the WNN structure in order to build a method to classify a collection containing a dataset of DNA sequences. This method is used to optimize an important number of inputs of DNA sequences. The Beta wavelet function is used to build the WNN. This wavelet makes the WNN training very efficient a reason of adjustable parameters of this function.

This paper contains five sections: in Sect. 2, we present our proposed approach. Section 2.6 presents the wavelet theory used to construct the WNN of our method. Section 3 shows the simulation results of our approach and Sect. 4 ends up with the conclusion.

2 Proposed Approach

This paper presents a new approach based on the wavelet neural network and Power Spectrum. The WNN is constructed by using the Multi-Library Wavelet Neural Networks (MLWNN). The WNN structure is solved by using the LTS method. The power spectrum is used to construct mathematical moments to solve the DNA sequence lengths. Our approach is divided into two stages: approximation of the input signal sequence and clustering of feature extraction of the DNA sequences using the WNN and the Euclidean distances is used to classify the feature extraction of the DNA sequences.

2.1 Fourier Transform and Power Spectrum Signal Processing

The proposed approach uses a natural representation of genomic data by binary indicator sequences of each nucleotide (adenine (A), cytosine (C), guanine (G), and thymine (T)). Afterwards, the discrete Fourier transform is used to these indicator sequences to calculate spectra of the nucleotides [11–20]. For example, if x[n] = [TT A A …], we obtain: x[n] = [000100011000 1000. ..]. The indicator sequence is manipulated with mathematical methods. The sequence of complex numbers, called f(x) (1), is obtained by using the discrete Fourier Transform:

$$f(x) = \sum_{n=0}^{N-1} X_e(n)\, e^{-j\pi k/N}, k = 0, 1, 2, \ldots N-1 \tag{1}$$

The Power Spectrum is applied to compute the Se[k] (2) for frequencies k = 0, 1, 2, …, N-1 is defined as,

$$Se[k] = |f(x)|^2 \tag{2}$$

Se [k] has been plotted (Fig. 1).

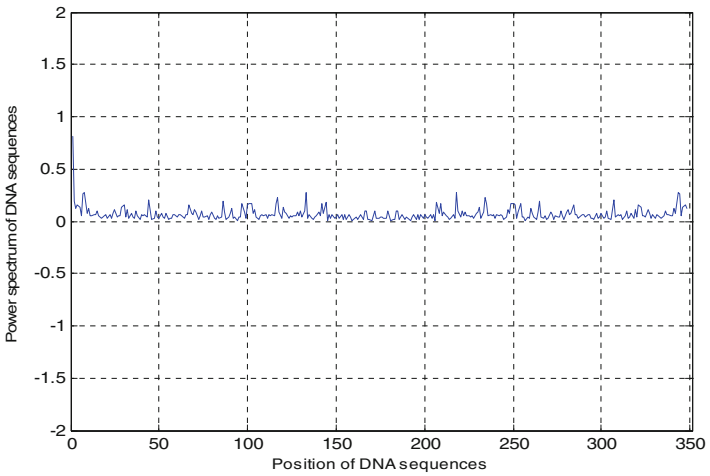


Fig. 1. Signal of a DNA sequence using Power Spectrum

2.2 Wavelet Neural Network

The wavelet neural network is defined by the combination of the wavelet transform and the artificial neuron networks [33, 34]. It is composed of three layers. The salaries of the weighted outputs are added. Each neuron is connected to the other following layer. The WNN (Fig. 2) is defined by pondering a set of wavelets dilated and translated from one wavelet candidate with weight values to approximate a given signal f. The response of the WNN is:

$$\hat{y} = \sum_{i=1}^{N_w} w_i \Psi\left(\frac{x - b_i}{a_i}\right) + \sum_{k=0}^{N_i} a_k x_k \tag{3}$$

where (x_1, x_2,…, x_{Ni}) is the vector of the input, Nw is the number of wavelets and y is the output of the network. The output can have a component refine in relation to the variables of coefficients a k (k = 0, 1… N_i) (Fig. 2). The wavelet mother is selected from the MLWNN, which is defined by dilation (a_i) which controls the scaling parameter and translation (b_i) which controls the position of a single function ($\Psi(x)$). A WNN is used to approximate an unknown function:

$$y = f(x) + \varepsilon \tag{4}$$

where f is the regression function and ε is the error term.

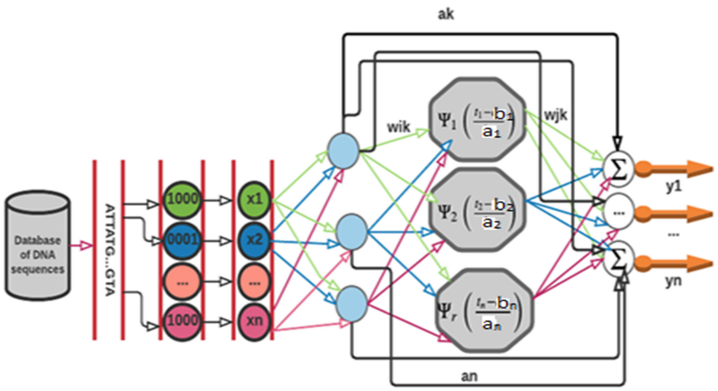


Fig. 2. The three layer wavelet network

2.3 Multi Library Wavelet Neural Network (MLWNN)

Many methods are used to construct the Wavelet Neural Network. Zhang applied two stages to construct the Wavelet neural Network [2, 3]. First, the discretely dilated and translated version of the wavelet mother function Ψ is used to build the MLWNN [21, 22].

$$W = \left\{ \psi_i : \psi_i(x) = \alpha_i \psi\left(\frac{(x_k - b_i)}{a_i}\right), \alpha_i = \left(\sum_{k=1}^{n}\left[\psi\left(\frac{(x_k - b_i)}{a_i}\right)\right]^2\right)^{\frac{1}{2}}, i = 1, \ldots L \right\}, \quad (5)$$

where L is the number of wavelets in W and x_k is the sampled input. Then the best M wavelet mother function is selected based on the training sets from the wavelet library W, in order to construct the regression:

$$f_M(x) = \hat{y} = \sum_{i \in I} w_i \psi_i(x), \quad (6)$$

where M $\leq$ L and I is a subset wavelet from the wavelet library.

Secondly, the minimized cost function:

$$j(I) = \min_{w_i, i \in I} \frac{1}{n} \sum_{k=1}^{n} \left(y_k - \sum_{i \in I} w_i \psi_i(x_k) \right)^2, \quad (7)$$

The gradient algorithms used to train the WNN, like least mean squares to reduce the mean-squared error:

$$j(w) = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}(w))^2, \quad (8)$$

where $j(w)$ is the output of the Wavelet neural networks. The time-frequency locality property of the wavelet is used to give a signal f , a candidate library w of wavelet basis can be constructed.

2.4 Wavelet Network Construction Using the LTS Method

The set of training data $TN = \{x_1, x_2, \ldots, x_k, f(x_k)\}_{k=1}^{N}$ is used to adjust the weights and the WNN parameters, and the output of the three layers of the WNN in Fig. 2 can be expressed via (7). The model selection is used to select the wavelet candidates from the Multi Library Wavelet Neural Networks (MLWNN). These wavelet mothers are used to construct the wavelet neural network structure [37, 38]. In this study, the Least Trimmed Squares estimator (LTS) is proposed to select a little subset of wavelet candidates from the MLWNN. These wavelet candidates are applied to construct the hidden layer of the WNN [30–32, 36]. Furthermore, the Gradient Algorithm is proposed to optimize the wavelet neural networks parameter. The residual (or error) e_i at the ith ouput of the WNN due to the ith example is defined by:

$$e_i = y_i - \hat{y}_i, i \in n \quad (9)$$

The Least Trimmed Square estimator is used to select the WNN weights that minimize the total sum of trimmed squared errors:

$$E_{total} = \frac{1}{2}\sum_{k=1}^{p}\sum_{i=1}^{l} e_{ik}^{2} \tag{10}$$

The Gradient Algorithm used to optimize the parameters (a_i,b_i,w_i) of the WNN.

2.5 Approximation of DNA Sequence Signal

The classification of DNA sequences is an NP-complete problem; the alignment is outside the range of two sequence of DNA, the problem rapidly becomes very complex because the space of alignment becomes very high. The recent advance of the sequence technology has brought about a consequent number of DNA sequences that can be analyzed. This analysis is used to determine the structure of the sequences in homogeneous groups using a criterion to be determined. In this paper, the Power Spectrum is used to process the signal of the DNA sequence. These signals are used by the wavelet neural networks (WNN) to extract the signatures of DNA sequences, which are used to match the DNA test with all the sequences in the training set [17–29]. Initially, the signatures of DNA sequences developed by the 1D wavelet network during the learning stage gave the wavelet coefficients which are used to adapt the DNA sequences test with all the sequences in the training set. Then, the DNA test sequence is transmitted onto the wavelet neural networks of the learning DNA sequences and the coefficients specific to this sequence are computed. Finally, the coefficients of the learning DNA sequences compared to the coefficients of the DNA test sequences by computing the Correlation Coefficient. In this stage, the Euclidean distances is used to classify the signatures of the DNA sequences [27].

The Euclidean distances of different DNA sequences are measured and applied as a measure of similarity for these DNA sequences. The pairwise Euclidean distances of DNA sequences are used to generate a similarity matrix, which can be used to classify the DNA sequence.

2.6 Learning Wavelet Network

In this section, we show how the library wavelet is used to learn a wavelet neural network [15, 16, 26, 27].

- Learning approach

Step 1: The data set of DNA sequence is divided into two groups: training and testing dataset. These groups are applied to train and test the wavelet neural network.
Step 2: Conversion of DNA sequence to a genomic signal using a binary indicator and Power Spectrum Signal Processing
Step 3: The discretely dilated and translated versions are used to construct the library W. The training data are proposed to create this library wavelet, apply the Least Trimmed

Square (LTS) algorithm to select the optimal mother wavelet function (10) (11) and choose, from the library, the N wavelet candidate that best matches an output vector.

Step3.1: Initializing of the mother wavelet function library

Step 3.2: Randomly initialize w_{jk} and V_{ij} .

Step 3.3: For k = 1,...,m

– Calculate the predicted output $\hat{y}_i$ via (3).
– Compute the residuals $e_{ik} = y_i - \hat{y}_i$ via (9).
The algorithm is stopped when the criteria diverged, then stop; otherwise, go to the next step.

– Find the arranged values $e_{ik}^2 \leq \ldots \leq e_{im}^2$. Choosing the N best mother wavelet function to initialize the WNN.

Step 4: The values of w_{ij}^{opt}, a_i^{opt} *and* b_i^{opt} are computed using the Gradient algorithm go to step 3.3.

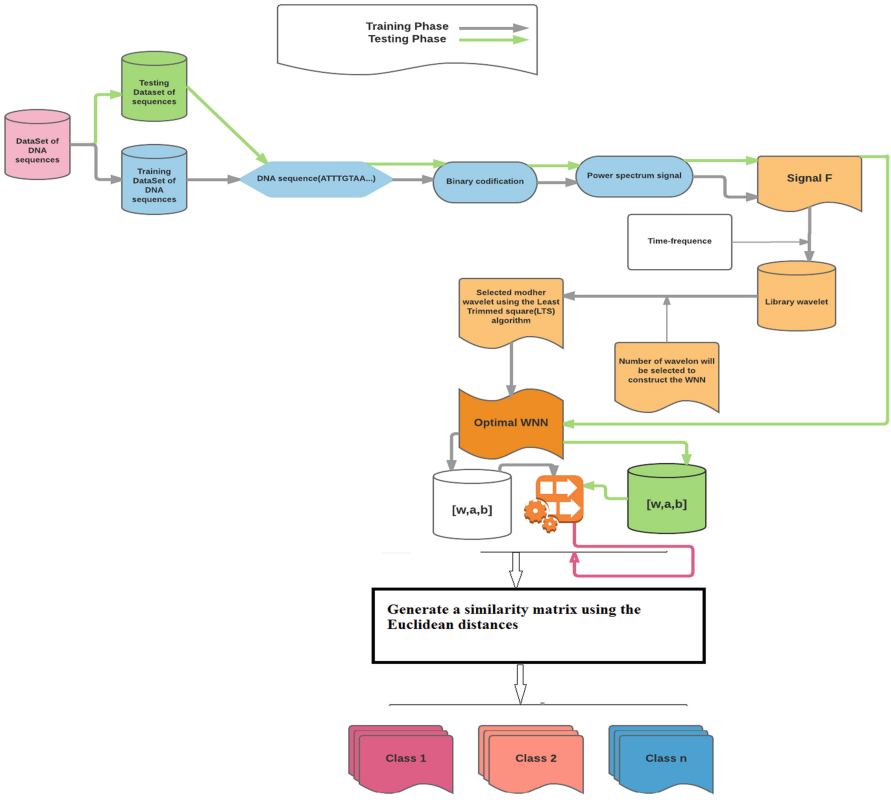


Fig. 3. Proposed approach

- Clustering using the Euclidean distances

  Step 1: Generate a similarity matrix, which can be used to classify the DNA sequence (w_{ij}^{opt}, a_i^{opt} and b_i^{opt}).
  Step 2: Use the similarity matrix to classify to the DNA sequence.

– To construct a phylogenetic tree of these sequences and Generate the classes of the DNA sequences. (The phylogenetic trees constructed from a similarity matrix reflect groups(classes) information, hierarchical similarity and evolutionary relationships of the DNA sequences) (Figure 3).

3 Results and Discussion

This paper used three datasets HOG100, HOG200, and HOG300 selected from microbial organisms [23]. In this study, different experiments are used to evaluate the performance of our approach. The data set of DNA sequences are divided into test and train data. The published empirical and synthetic datasets are selected to perform the clustering comparative analysis [23] (Table 1).

Table 1. Distribution of available data into training and testing set of DNA sequence

Dataset	Total	Training	Test
HOG100	500	300	200
HOG200	600	400	200
HOG300	700	600	100

3.1 Classification Results

Experiment results were performed to prove the effectiveness of our proposed approach. Evaluation metrics namely Precision, Recall and accuracy are used to compare our approach with other competitive methods. The classification accuracy Ai of an individual program i depends on the number of samples correctly classified (true positives plus true negatives) and is evaluated by the formula:

$$A_i = \frac{t}{n} * 100 \tag{11}$$

where t is the number of sample cases correctly classified, and n is the total number of sample cases.

Table 2 and Fig. 4 show that WNN-PS (our method) is better than other models (WFV, K-tuple and DMK) in terms of the classification results and optimal settings. The number of classes obtained by our approach is little less than in the other methods.

Table 2. The classification results of WNN- PS(Our Method) and other models (WFV, K-tuple, DMK) on different datasets of DNA sequences

	Our method		WFV		K-tuple		DMK	
Dataset	Accuracy (%)	# class	Accuracy (%)	#class	Accuracy (%)	# class	Accuracy (%)	# class
HOG100	**98.23**	**330**	57.25	854	56.36	451	58.25	658
HOG200	**88.54**	**468**	66.25	845	53.55	566	66.69	754
HOG300	**97.77**	**232**	58.68	185	62.36	654	71.36	256

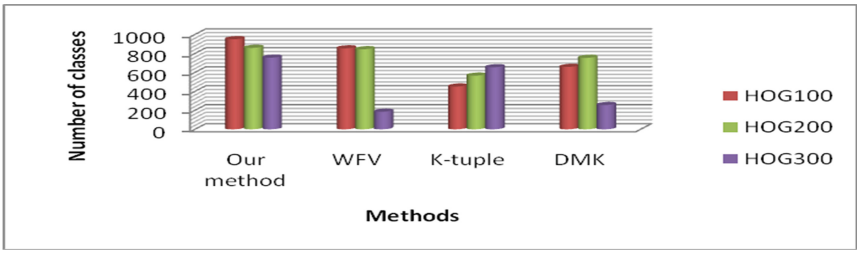


Fig. 4. The number of classes obtained using the proposed approach and the other models

The accuracy proves the efficiency of our method. The accuracy is increased using WNN and LTS method. The LTS is applied to optimize the WNN structure.

3.2 Running Time

Tables 2 and 3 show that the WNN can produce very good the prediction accuracy. The results of our approach WNN-PS tested on datasets show that accuracy outperforms the other techniques in terms of percentage of the correct species identification. Tables 2 and 3 show the distribution of the good classifications by class as well as the rate of global classification for all the DNA sequences of the validation phase. The WNN-PS(our approach) is faster than the other methods. This speed is due to the use of the Least Trimmed Square (LTS) algorithm; this method is a robust estimator.

Table 3. Running time in seconds of each method on all datasets

Dataset	Model	Length of feature vector	Total running time
HOG100	**Our Method**	**128**	**75.1254**
	WFV	32	110.7491
	K-tuple	64	771.4767
HOG200	**Our Method**	**128**	**224.325**
	WFV	32	666.3615
	K-tuple	64	3030.1732
HOG300	**Our Method**	**128**	**780.457**
	WFV	32	1373.7718
	K-tuple	64	5582.9042

4 Conclusions

In this study, we have used the LTS method to select a subset of wavelet function from the Library Wavelet Neural Network Model. This subset wavelet is applied to build Wavelet Neural Network (WNN). The WNN is used to approximate function f (x) of a DNA sequence signal. Firstly, the binary codification and Power Spectrum are used to process the DNA sequence signal. Secondly, the Library Wavelet is constructed. The LTS method is used to select the best wavelet from library. These wavelets are applied to construct the WNN. Thirdly, the Euclidean distances of signatures of DNA are used to classify the similar DNA sequences according to some criteria. This clustering aims at distributing DNA sequences characterized by p variables X1, X2… Xp in a number m of subgroups which are homogeneous as much as possible while every group is well differentiated from the others. The proposed approach helps to classify DNA sequences of organisms into many classes. These clusters can be used to extract significant biological knowledge.

References

1. Zhang, Q., Benveniste, A.: Wavelet networks. IEEE Trans. Neural Networks **3**(6), 889–898 (1992)
2. Zhang, J., Walter, G., Miao, Y., et al.: Wavelet neural networks for function learning. IEEE Trans. Signal Process. **43**(6), 1485–1497 (1995)
3. Zhang, Q.: Using wavelet network in nonparametric estimation. IEEE Trans. Signal Process. **8**, 227–236 (1997)
4. Pati, Y.C., Krishnaprasad, P.S.: Analysis and synthesis of feed-forward neural networks using discrete affine wavelet transformations. IEEE Trans. Neural Networks **4**, 73–85 (1993)
5. Billings, S.A., Wei, H.L.: A new class of wavelet networks for nonlinear system identification. IEEE Trans. Neural Networks **16**, 862–874 (2005)
6. Xu, J.H., Ho, D.W.C.: A basis selection algorithm for wavelet neural networks. Neurocomputing **48**, 681–689 (2002)
7. Mallat, S.G., Zhifeng, Z.: Matching pursuits with time-frequency dictionaries. IEEE Trans. Signal Process. **41**, 3397–3415 (1993)
8. Chen, S., Wigger, J.: Fast orthogonal least squares algorithm for efficient subset model selection. IEEE Trans. Signal Process. **43**, 1713–1715 (1995)
9. Han, M., Yin, J.: The hidden neurons selection of the wavelet networks using support vector machines and ridge regression. Neurocomputing **72**, 471–479 (2008)
10. Wu, C.H.: Artificial neural networks for molecular sequence analysis. Comput. Chem. **21**(4), 231–256 (1997)
11. Jach, E.A., Marín, J.M.: Classification of genomic sequences via wavelet variance and a self-organizing map with an application to mitochondrial DNA. Stat. Appl. Genet. Mol. Biol. **9**, 1544–6115 (2010)
12. Zhao, J., Yang, X.W., Li, J.P., Tang, Y.Y.: DNA sequences classification based on wavelet packet analysis. In: Tang, Y.Y., Yuen, P.C., Li, C.-H., Wickerhauser, V. (eds.) WAA 2001. LNCS, vol. 2251, pp. 424–429. Springer, Heidelberg (2001). doi:10.1007/3-540-45333-4_53

13. Wu, C., Berry, M., Fung, Y.-S., McLarty, J.: Neural Networks for Molecular Sequence Classification. In: Proceedings of the International Conference on Intelligent Systems for Molecular Biology, pp. 429–437 (1993)
14. Vinga, S., Almeida, J.: Alignment-free sequence comparison-a review. Bioinformatics **19**, 513–523 (2003)
15. Wei, D., Jiang, Q.: A DNA sequence distance measure approach for phylogenetic tree construction. In: Proceedings of the IEEE Fifth International Conference on Bio-Inspired Computing: Theories and Applications (BIC-TA), pp. 204–212, IEEE (2010)
16. Shi, L., Huang, H.: DNA sequences analysis based on classifications of nucleotide bases. In: Luo, J. (ed.) Affective Computing and Intelligent Interaction. AISC, vol. 137, pp. 379–384. Springer, Heidelberg (2012). doi:10.1007/978-3-642-27866-2_45
17. Bauer, M., Schuster, S.M., Sayood, K.: The average mutual information profile as a genomic signature. BMC Bioinform. **9**, 48 (2008)
18. Qi, J., Wang, B., Hao, B.I.: Whole proteome prokaryote phylogeny without sequence alignment: a K-string composition approach, vol. 58, pp 1–11 (2004)
19. Bonham-Carter, O., et al.: Alignment-free genetic sequence comparisons: a review of recent approaches by word analysis. Brief. Bioinform. **15**(6), 890–905 (2013)
20. Bao, J.P., Yuan, R.Y.: A wavelet-based feature vector model for DNA clustering. Genet. Mol. Res. **14**, 19163–19172 (2015)
21. Amar, C.B., Bellil, W., Alimi, M.A.: Beta function and its derivatives: a new wavelet family. Trans. Syst. Signals Devices **1**, 275–293 (2006)
22. Bellil, W., Othmani, M., Amar, C.B.: Initialization by selection for multi-library wavelet neural network training. In: Conference: Artificial Neural Networks and Intelligent information Processing (ANNIIP), Angers, France (2007)
23. http://doua.prabi.fr/databases/hogenom/
24. Mejdoub, M., Amar, C.B.: Classification improvement of local feature vectors over the KNN algorithm. Multimedia Tools Appl. **64**(1), 197–218 (2013)
25. Zaied, M., Said, S., Jemai, O., Amar, C.: A novel approach for face recognition based on fast learning algorithm and wavelet network theory. Int. J. Wavelets Multiresolut. Inf. Process. **19**, 923–945 (2011). World Scientific
26. Said, S., Amor, B.B., Zaied, M., Amar, C.B., Daoudi, M.: Fast and efficient 3D face recognition using wavelet networks. In: 16th IEEE International Conference on Image Processing, Cairo, Egypt, pp. 4153–4156 (2009)
27. Jemai, O., Zaied, M., Amar, C.B.: Fast learning algorithm of wavelet network based on fast wavelet transform. Int. J. Pattern Recogn. Artif. Intell. **25**(8), 1297–1319 (2011)
28. Jemai, O., Zaied, M., Amar, C.B., Alimi, A.: Pyramidal hybrid approach: wavelet network with OLS algorithm- based image classification. Int. J. Wavelets Multiresolut. Inf. Process. **9**, 111–130 (2011). World Scientific Publishing Company
29. Ejbali, R., Benayed, Y., Zaied, M., Alimi, A.: Wavelet networks for phonemes recognition. International Conference on Systems and Information Processing (2009)
30. Ejbali, R., Zaied, M., Amar, C.B.: Multi-input Multi-output Beta wavelet network modeling of acoustic units for speech recognition. Int. J. Adv. Comput. Sci. Appl. (IJACSA), The Science and Information Organization(SAI), vol. 3 (2012)
31. Ejbali, R., Zaied, M., Amar, C.B.: Wavelet network for recognition system of arabic word. Int. J. Speech Technol. **13**, 163–174 (2010). Springer edition
32. Bouchrika, T., Zaied, M., Jemai, O., Amar, C.B.: Ordering computers by hand gestures recognition Based on wavelet networks. In: International Conference on Communications, Computing and Control Applications, Marseilles, France, pp. 36–41 (2012)

33. Mejdoub, M., Fonteles, L., Amar, C.B., Antonini, M.: Embedded lattices tree: an efficient indexing scheme for content based retrieval on image databases. J. Vis. Commun. Image Represent. **20**(2), 145–156 (2009)
34. Dammak, M., Mejdoub, M., Zaied, M., Amar, C.B.: Feature vector approximation based on wavelet network. In: Proceedings of the 4th International Conference on Agents and Artificial Intelligence (ICAART 2012), vol. 1, pp. 394–399

A Spiking Neural Network Model with Fuzzy Learning Rate Application for Complex Handwriting Movements Generation

Mahmoud Ltaief(✉), Hala Bezine, and Adel M. Alimi

REGIM-Lab.: REsearch Groups in Intelligent Machines,
University of Sfax, ENIS, BP 1173, 3038 Sfax, Tunisia
mahmoud.ltaief@ieee.org

Abstract. In this paper a spiking neural network model with fuzzy learning rate for online complex handwriting movement generation is proposed. The network is composed of an input layer which uses a set of Beta-elliptic parameters as input, a hidden layer and an output layer dealing with the estimation of the script coordinates $X(t)$ and $Y(t)$. An additional input is used as a timing network to prepare the input parameters. We also propose a Fuzzy Learning Rate (FLR) for our spiking neural network. This rate is obtained by combining an Adaptive Learning Rate (ALR) with a fuzzy logic based supervisor. The obtained results showed the efficiency of the proposed fuzzy strategy for the online adjustment of the learning rate. Indeed, we have improved, indifferently from the initialization, the Neural Network training quality in terms of rapidity and precision. Similarity degree is measured between original and generated scripts to evaluate our model.

Keywords: Online handwriting · Beta-elliptic model · Spiking neural network · Fuzzy Learning Rate

1 Introduction

Writing is considered among the fastest motor activity and the most complex of our directory engine. This activity applies a coordination of several joints and muscles to generate a succession of graphic shapes quickly and suitably precise to be known [12]. The writer begins with the intention of writing a message (semantic level), that transforms into words (syntactical and lexical level). When the single letters (graphemes) are recognized, the writer chooses the specific letter shape (allograph). The choice is based on selection syntax of formal allograph, random selection or personal preference [10,11]. Then, allographs are converted into movement patterns, that is the subject of this work. The major contributions of this work are: the use of the Beta-elliptic model for parameters extraction giving a full description in kinematic and static domain for online handwriting. Secondly the use of this parameters given by the Beta elliptic model as input to

A. Abraham et al. (eds.), *Proceedings of the 16th International Conference on Hybrid Intelligent Systems (HIS 2016)*, Advances in Intelligent Systems and Computing 552,
DOI 10.1007/978-3-319-52941-7_40

our spiking neural network which have the advantage of providing a temporal integration of information to generate the pen tip coordinates $x(t)$ and $y(t)$ unlike the Gangadhar model that produces velocities ($V_x(t)$, $V_y(t)$) and then with a simple mathematical operation it produces $x(t)$ and $y(t)$. The outline of the paper is as follows: in Sect. 2, we present our spiking model for online handwriting generation which contain three parts. In the first part we present the Beta elliptic model for handwriting modeling and parameters extraction. In the second part we present our spiking neural network. In the third part we detail the fuzzy learning rate. In Sect. 3, simulation results for model validation are presented. We finish this paper by a conclusion.

2 The Proposed Model

The suggested model is concerned to online handwriting generation and it is inspired from the Beta-elliptic model for complex handwriting movements modeling. In our model, a handwritten stroke is approximated in the dynamic domain by a Beta profile which in turn corresponds to an elliptical arc in the trajectory domain. Also, a stroke runs from random starting position is typified by ten parameters. The first five parameters concern neuromuscular networks synchronization properties involved in the movement generation, although the final five parameters depict the geometrical properties of the generated trace. As illustrated in Fig. 1, these parameters are used as input for the spiking neural network to produce the initial script. A fuzzy supervisor is developed to improve the learning phase as well in term of accuracy and speed.

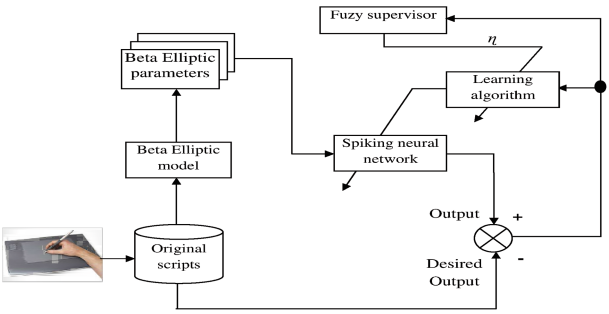


Fig. 1. The architecture of the handwriting generation model.

2.1 The Beta-elliptic Model for Complex Handwriting Movements Modelling

The beta-elliptic model is founded on certain hypothesis: first of all, it supposes handwriting movement like all other extremely qualified motor process is partly programmed beforehand. Secondly, it assumes that movements are planned and

represented in the velocity field since the invariant most accepted in handwriting movement generation is the beta function for velocity profile modeling. In its most simple form, the model is funded on a beta equation $\beta(t, t_0, t_1, t_c, p, q)$, where t_0 is the beginning time, t_1 is the ending time, p and q are intermediate parameters. This equation represents in the kinematics domain the velocity profile which is in turn described by an elliptic arc which typify the trajectory in the static domain [1–3]. For an elementary movement named stroke the Beta-elliptic model produces a set of ten parameters that characterizes the movement in both fields static and kinematic. Then for each stroke we have ten parameters (t_c, p, q, t_0, t_1, a, b, x_0, y_0, θ). These parameters will be used as input for our spiking neural network for complex handwriting movements generation [6,7].

2.2 The Spiking Neural Network Model

Spiking neural networks (SNN) is a particular category of artificial neural networks (ANN), in which neurons intercommunicate by sequences of spikes. Spiking neural networks are able to handle large quantity of data using relatively a few number of spikes [13]. Because of their functional likeness to biological neurons, spiking models provide efficient tools for analyzing the basic processes in the brain, comprising neural plasticity, learning and information processing.

Spiking neural networks are designed to model more closely the human observed behavior and neuron. There are many parts that are computationally important; the neuron itself, the synapses connecting the neurons and the signal or spike which moves between neurons through the synapses [9,13].

The spiking neural network architecture. The architecture of our network consists of a feed-forward spiking neural network see Fig. 2. The input layer is composed of 11 neurones, the first 10 neurones are the Beta-elliptic parameters (the static parameters given by elliptic equation and the dynamic parameters given by Beta equation). Also we use one neurone for the timing network [8].

The timing network is used as a synchronizer between the input and output of our network: in the parameters extraction stage every stroke was described with a set of beta profile and set of elliptic arc. Each couple (beta profile, elliptic arc) belongs to a time interval $[t_0,t_1]$. The timing network duplicates the parameters given by each pair (Beta profile, elliptic arc) in the appropriate time interval. Duplication is done so that each point $(x(t), y(t))$ of the network output corresponds to an input (the beta elliptic parameters) at time t. In our neural network we use one hidden layer with 10 neurones (7 excitatory neurones and 3 inhibitory neurones). Finally the output layer contain 2 neurones that represent the coordinates of script en tip $(x(t), y(t))$ estimated by the network. In our spiking neural network every two interconnected neurons have a number of m=8 synaptic terminals with variant delays d^k and an individual weight Fig. 2(B). Between the firing time of the pre-synaptic neuron, the post-synaptic potential begins to rise or fall if the neuron is excitatory, or inhibitory. Inhibitory neurons generate only negative post-synaptic potentials [4]. The neuron activity is described by the Spike Response Model [5], describing the neuron with one

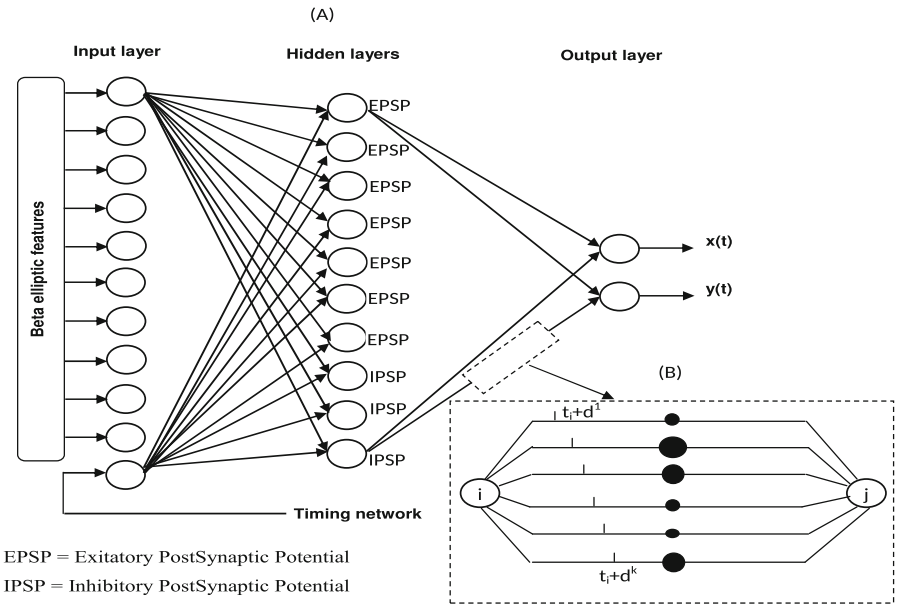


Fig. 2. (A) The spiking neural network architecture. (B) connection consisting of multiple delayed synaptic terminals.

variable, the membrane potential $x\,(t)$. A neuron j receives input from a set of pre-synaptic neurons $i \in \Gamma_j$ is characterized by the variable $x_j(t)$. In that case, the effects of the arrived spikes are summed. If this sum (noted $x_j(t)$) achieves the threshold v a potential action is triggered (see Fig. 3). The membrane potential of one neuron j is described with the equation below:

$$x_j\,(t) = \sum_{i \in \tau_j} \sum_{k} w_{ij}^k y_i^k\,(t) = \sum_{i \in \tau_j} \sum_{k} w_{ij}^k \varepsilon\left(t - t_i - d^k\right) \tag{1}$$

Where w_{ij}^k is the synaptic weight. The sum covers every presynaptic spikes where Γ_j is the totality of the presynaptic neurons of neuron j, and t_i is the firing time of presynaptic neuron $i \in \Gamma_j$. The term w_{ij}^k represents the weight of the synaptic terminal k, between the neurons j and i, having the delay d_k.

$$E = \frac{1}{2} \sum_{j \in J} \left(t_j^a - t_j^d\right) \tag{2}$$

In this paper Error-backpropagation learning algorithm is used to update each synaptic weight w_{ij}^k to reduce the error E. The E variation rate with respect to w_{ij}^k must be known. The back-propagation algorithm derived from the same way as in [8].

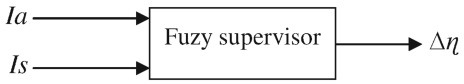


Fig. 3. Functional schema of the fuzzy supervisor.

2.3 Fuzzy Learning Rate

The limit of the backpropagation learning algorithm
The main disadvantage of the backpropagation algorithm is the slowness in convergence, mainly in the case of a large network (having a large number of weight adjusting). Furthermore, in addition to its mandatory convergence to a global minimum, learning by backpropagation algorithm depends a lot on the initialization of the connection weights and the size of the training set. However, the initialization of weights with small random values and the use of a relatively large number of examples can lead to satisfactory results. In addition to their dependence to initializing and to the characteristics of the examples base (richness and size), the convergence properties of the backpropagation algorithm is closely related to the value of the learning rate. Indeed, a constant value of η may not be suitable for every iteration and does not, therefore, lead in all cases, to the minimization of error and to sufficient rapidity learning. Indeed, a relatively small constant value may freeze the training procedure in areas of gentle slope and makes, therefore, very slow convergence rate. On the other hand, accelerating the convergence at the cost of an oscillatory criterion can be obtained by a choice of a relatively large constant value of η. Considering the desirability of the learning rate of the backpropagation algorithm to improve the quality of learning in multilayer networks and to the good adaptation of the fuzzy logic expression of imprecise conceptsis proposed combining a supervisor including fuzzy logic with backpropagation algorithm for adaptive learning rate. The resulting algorithm is called "backpropagation algorithm fuzzy learning coefficient" and as the functional diagram that of Fig. 1. For example, the proposed scheme is on spiking neural network model for handwriting movement generation. In the following section of this article, we introduce the fuzzy learning rate. The results, revealing the effectiveness of the proposed fuzzy control strategy, the subject of Sect. 3. In Sect. 4, a conclusion completes this article. *Fuzzy supervisor* For online learning rate adjustment using fuzzy logic, we opted for a fuzzy supervisor which, by integrating into the train loop of the backpropagation algorithm, adapts the learning rate η in the sense of performance criterion on the speed and accuracy of learning. We must therefore inquire at each iteration on the quality of learning through indices of accuracy Ia and speed Is, the supervisor have two inputs and one output (variation of η: $\Delta\eta$) as in Fig. 3.

a- Supervisor inputs

– Index of accuracy

It is obvious that the index of accuracy can be quantified by the normalized difference between the current sum of the learning squared errors Seq (K) and the error goal Eg. The index may be, for example, formulated at each iteration as follows:

$$Ia(k) = \left(\frac{Seq(k) - Eg}{Seq(1) - Eg}\right)) * 100 \tag{3}$$

– Index of speed

The normalized difference between the current and previous sums of squared errors can learn about the convergence and its speed. Therefore, it can be considered, in the following formulation, as speed index:

$$Is(k) = \left(\frac{Seq(k) - Seq(k-1)}{Seq(1) - Eg}\right)) * 100 \tag{4}$$

The fuzzy supervisor implements three main functions; namely the fuzzification, fuzzy inference and defuzzification [BEN 01].

b- Fuzzification

The fuzzification is to convert the normalized real inputs (Ia_n, Is_n) into fuzzy inputs. This can be done by combining a set of triangular membership functions centered on the numbers Nci_j to the ni considered linguistic variables. If ni represents the number of fuzzy sets associated with each input, the normalized variables Ia_n and Is_n and central numbers are written in this case as follows:

– For the accuracy index, the normalization interval is [0, 1]

$$Ia_n(k) = \frac{Ia(k) - Ia_{min}}{Ia_{max} - Ia_{min}} \tag{5}$$

$$Ncia_j = \frac{j-1}{ni-1} \tag{6}$$

With: $1 \leq \mathrm{j} \leq \mathrm{ni}$.

– For the speed index, the normalization interval is [−1, 1].

$$Is_n(k) = \frac{Is(k) - \frac{Is_{max}+Is_{min}}{2}}{\frac{Is_{max}-Is_{min}}{2}} \tag{7}$$

$$Ncis_i = -1 + \frac{i-1}{\frac{ni-1}{2}} \tag{8}$$

With: $1 \leq \mathrm{i} \leq \mathrm{ni}$.

In the present case, three fuzzy sets have been reserved for each supervision input ($ni = 3$) represented by triangular membership functions as illustrated by the Fig. 4.

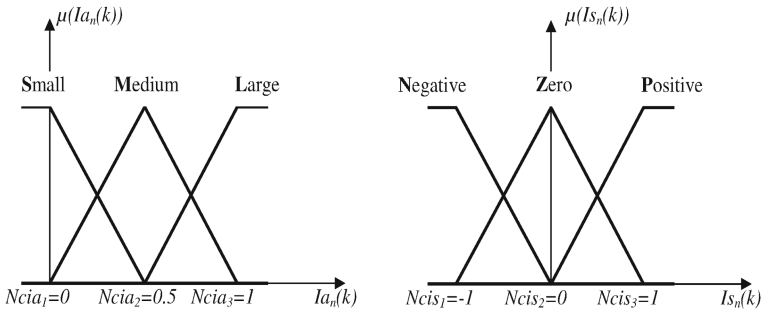


Fig. 4. Membership functions related to the supervision inputs.

Table 1. Table of inference.

$\Delta\eta$		Ia_n		
		S	M	L
Is_n	N	PS	P	PL
	Z	Z	PS	P
	P	NS	N	NL

c- Fuzzy inference

The inference step is to apply linguistic rules provided by an established base prior on the fuzzy inputs from the fuzzification for evaluating the output of supervisor ($\Delta\eta$). The inference table (Table 1) is established by appeal to the decision logic considered in the case of adaptive learning rate and by using some experimental trials evaluating criterion on the sum of squared errors in the case of a spiking neural network for handwriting generation. This criterion highlights the influence of the choice of the rate η on the learning quality of the gradient backpropagation algorithm. Indeed, the choice of a relatively high learning rate saves on speed at the expense of insufficient precision. the decrease in the value of η under the same conditions allows, on the other hand improving the accuracy at the cost of a loss on the speed of learning. The linguistic variables are given:

– For $Ia_n(k)$: S: Small; M: Medium; L: Large.
– For $Is_n(k)$: N: Negative; Z: Zero; P: Positive.

The conditions are expressed by linguistic terms describing the inputs while the condition is expressed by the linguistic terms describing the supervision output. The inference table imposes no fuzzy sets (FS1, ..., FSno with no = 7) for the output variable $\Delta\eta$. As in the case of inputs, triangular membership functions were retained for the supervision output. These functions are defined on the normalized interval [−1, 1] and focused on the Nco_l numbers.

$$Nco_l = -1 + \frac{l-1}{\frac{no-1}{2}} \tag{9}$$

With: $1 \leq l \leq$ no.

The MAX-MIN inference method is used to evaluate the contribution of rules. This method sets the logical operation "AND" to a minimum and the logic operation "OR" to a maximum. Therefore, the possible output by the supervisor is determined by computing the Dl term given by the following formula:

$$D_l = \underset{Rule\ l}{MAX} \{MIN\,[\mu_i(Ia_n), \mu_j(Is_n)]\} \tag{10}$$

With: $i, j = 1, \ldots, ni$ and $l = 1, \ldots, no$. In fact, the terms D_l correspond to the heights of the trapezoids obtained by capping of triangular membership functions.

d- Defuzzification

The defuzzification operation, which involves calculating a numerical value of the output of supervision from its fuzzy value, is the final step of supervision. This value ($\Delta\eta$) can be calculated by the barycenter maxima method whose expression is:

$$\Delta\eta = \frac{\sum_{l=1}^{no} D_l * Nco_l}{\sum_{l=1}^{no} D_l} \tag{11}$$

Finally, the effective output η that will serve as a learning factor is calculated for each iteration K depending $\Delta\eta$ as follows:

$$\eta(k) = \eta(k-1) + g\Delta\eta(k) \tag{12}$$

With g is a positive weighting. This weighting is that of influence on how quickly we will reach the optimal learning rate within the meaning of the established decision rules. The choice of g may be practically based on the held range for the variation of η (for example the tenth of the range). If one opts for a learning rate between 0 and 1, a weight of 0.1 may then be suitable for different situations.

3 Simulation Results

To validate the intake of performance of the fuzzy learning rate so developed, it was considered, by way of example, a spiking neural network with fuzzy learning rate for handwriting generation. The simulation conditions can be summarized as follows: the architecture and the input of our spiking neural network is as defined in Sects. 2.1 and 2.2. To train our spiking neural network 1000 scripts (digits, Latin and Arabic words and letters) are used for learning and 500 scripts for testing. The Fig. 5 shows that the fuzzy supervisor find always the adequate learning rate according to the established rule base. Indeed, the sum of squared errors recorded in the presence of fuzzy supervisor, were quite low compared to case where the constant learning rate (CLR) was adopted. This shows the effectiveness of fuzzy strategy applied to improve the quality of learning. Figure 6 shows the simulation results: generated scripts when using CLR (B) and generated scripts using FLR (C) compared with original scripts (A). Refering to this figure we can note that scripts generated when using FLR are better and more representative of the original scripts. this is confirmed by the measure of

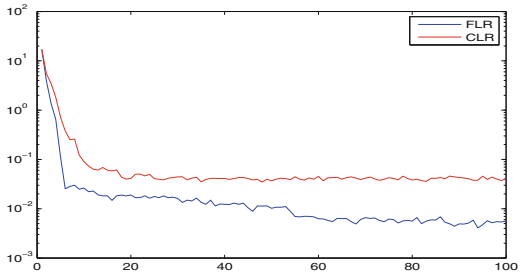


Fig. 5. The training learning error using the fuzzy learning rate (FLR) and the constant learning rate (CLR).

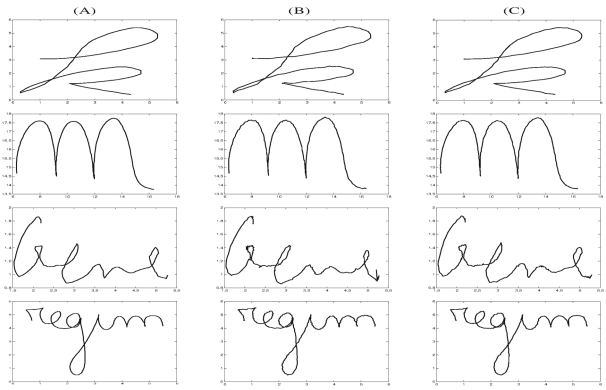


Fig. 6. (A) the original scripts (B) the scripts generated with CLR and (C) the scripts generated with FLR.

the similarity degrees across the entire database. Indeed the similarity degree between the original scripts and the generated scripts in the case of CLR is in the order of 0.85 however in the case of the FLR the similarity degree is in the order of 09.

4 Conclusion

The sensitivity of the learning quality of the learning rate choice prompted us to propose a new online strategy of rate adjustment. This strategy is based on the determination of the learning rate by a fuzzy supervisor. The fuzzy approach developed favors obtaining a good quality of learning in terms of speed and precision, compared to the approach using constant learning rate. We note, finally, a good choice of universe of discourse for the inputs and output of the supervisor should lead to improved results.

References

1. Alimi, M.A.: Beta neuro-fuzzy systems. TASK Q. J. **7**(1), 23–41 (2003). Special Issue on "Neural Networks" edited by W. Duch and D. Rutkowska
2. Bezine, H., Alimi, M.A., Sherkat, N.: Generation and analysis of handwriting script with the Beta-elliptic model. In: Proceedings of the 9th International Workshop on Frontiers in Handwriting Recognition IWFHR 2004, Tokyo, Japan, pp. 515–520 (2004)
3. Bezine, H., Kefi, M., Alimi, M.A.: On the Beta-elliptic model for the control of human arm movements. IJPRAI **21**(1), 5–19 (2007)
4. Bohte, S., Kok, J., Poutré, H.L.: Error backpropagation in temporally encoded networks of spiking neurons. Neurocomputing **48**, 17–37 (2002)
5. Gerstner, W.: A framework for spiking neuron models: the spike response model. In: Moss, F., Gielen, S. (eds.) The Handbook of Biological Physics, vol. 4, pp. 469–516. Elsevier Science (2001). Chap. 12
6. Ltaief, M., Njah, S., Bezine, H., Alimi, M.A.: Genetic algorithms for perceptual codes extraction. J. Intell. Learn. Syst. Appl. JILSA **4**, 256–265 (2012)
7. Ltaief, M., Bezine, H., Alimi, M.A.: A neuro-Beta-elliptic model for handwriting generation movements. In: International Conference on Frontiers in Handwriting Recognition ICFHR, pp. 799–804 (2012)
8. Ltaief, M., Bezine, H., Alimi, M.A.: A spiking neural network model for complex handwriting movements generation. Int. J. Comput. Sci. Inf. Secur. (IJCSIS) **14**(7), 319–327 (2016)
9. Natschlaeger, T., Maass, W.: Spiking neurons and the induction of finite state machines. Theor. Comput. Sci. Spec. Issue Nat. Comput. **287**, 251–265 (2002)
10. Schomaker, L.R.B.: From handwriting analysis to pen-computer applications. Electron. Commun. Eng. J. **10**(3), 93–102 (1998)
11. Schomaker, L.R.B.: Simulation and recognition of handwriting movements: a vertical approach to modeling human motor behavior. Ph.D. thesis. Nijmegen University, Netherlands (1991)
12. Teulings, H.L., Thomassen, A., Schomaker, L.R.B., Morasso, P.: Experimental protocol for cursive script acquisition: the use of motor information for the automatic recognition of cursive script. Report 3.1.2., ESPRIT Project, 419 (1986)
13. VanRullen, R., Guyonneau, R., Thorpe, S.J.: Spike times make sense. Trends Neurosci. **28**, 1–4 (2005)

A Method Proposed for Estimating Depressed Feeling Tendencies of Social Media Users Utilizing Their Data

Marouane Birjali[1(✉)], Abderrahim Beni-Hssane[1], and Mohammed Erritali[2]

[1] LAROSERI Laboratory, Department of Computer Sciences, Faculty of Sciences, University of Chouaib Doukkali, El Jadida, Morocco
birjali.marouane@gmail.com, abenihssane@yahoo.fr
[2] TIAD Laboratory, Department of Computer Sciences, Faculty of Sciences and Technologies, University of Sultan Moulay Slimane, Béni Mellal, Morocco
m.erritali@usms.ma

Abstract. The enormous progress of social networks and the large amount of data generated by them, has led many studies the possibility to identify the hidden knowledge. Depression is more than just feeling unhappy or fed for a few days and affects people in different ways and can cause a variety of symptoms. In this paper, we aim to analysis on research for feelings of depression for user's activities in Twitter, which is a popular microblogging site, for estimating his/her depressive tendency. Then we investigate Weka as a tool of machine learning classification to extract useful information for classification of Twitter Data collected from Twitter based Twitter API. Therefore, we perform our experiments to estimate participants depressive tendencies using an algorithm for computing the semantic similarity between tweets in training set and data set based on WordNet as an external semantic resource. Experimental results show that Twitter could be used to analyze online depression feeling. In addition, this study demonstrated that we can extract sentiments of Twitter users from social networks.

Keywords: Depression feeling · Machine learning · Twitter API · Semantic · Sentiment analysis

1 Introduction

Recently, there have been some outstanding social media analytic works, especially on the topic Twitter opinion analysis. Most of these attempts to extract public thoughts and opinions are based on sentiment mining techniques. The social network is more than ever a means of communication tools for information exchange. It offers a considerable information at a great speed and services adapt more to the needs of users. These can consult the Internet to create social communications, social interaction (between individuals or groups of individuals), and content creation [1].

During the last years, the Internet has yet seen a wider scope through the development of social media. Based on easy communication techniques and accessible to all, the media promote social interaction through the Internet. Offering free access, social

A. Abraham et al. (eds.), *Proceedings of the 16th International Conference on Hybrid Intelligent Systems (HIS 2016)*, Advances in Intelligent Systems and Computing 552,
DOI 10.1007/978-3-319-52941-7_41

media has greatly promoted the mass and have triggered public debate on the Internet. Many social networks exist and there are more than 900 social media sites available on the internet [2]. Millions of people are using Twitter and it is ranked as one of the most visited sites with the average of 58 million tweets per day [3].

"Tweet" is short post on Twitter. Tweets are created in real time. With the shortness guaranteed in 140- character limit of the posts and the popularity of mobile applications Twitter, the tweet and retweet users instantly. See the information diffused quickly in a social network. For example, crime associated with the shooting can be detected in twitter in just 10 min following the incident took place, but it takes about 3 h to the news report [4].

In this work, social network like Twitter and Facebook are increasingly associated with phenomena such as harassment, bullying or even depression [5]. It is therefore very important to detect potential victims at the earliest in order to strengthen depression prevention on the web. Indeed, we can cite as an example the case of two American rappers Freddy E. [6] and Capital Steez [7] who are given the death commenting live on their actions their Twitter accounts.

This paper is organized as follows. Some related work on topic model in Sect. 2 and our Research Background of work is presented in Sect. 3. Section 4 describes experiments setup and the result. Finally, the paper is summarized briefly in Sect. 5.

2 Related Works

The social network has attracted the attention of the research community that is trying to understand, among others, their structure and user interconnection and interaction between users. People tend to express their feelings and talk about their activities of daily life through Twitter.

Applying machine learning methods for the identification of depression has grown in recent years. Linguistic Inquiry and Word Count Version 2007 (LIWC2007) evaluates different words or emotional, cognitive and structural English expressions presented in oral and written sentences of individuals. Liwc could be used to identify the trend of an emotional post. Ramirez-Esparza, et al. [8] worked on linguistic markers and for discussion of depression by gathering information both in depressed individuals and non-depressed existing Internet forums using bulletin board systems (BBS). They also found that depressed people who wrote in English were more likely to report medical problems.

Sentiment analysis was processed as a natural language processing task on many levels of granularity. There have been a broad range of research on the felt-ment analysis [9] from rule base, bag-of-words machine learning methods approaches. From being a task of classification at record level Turney [10], he has been treated at the sentence Hu and Liu [11] and more recently at the Wilson et al. sentence [12].

The social network like Twitter, on which users post his reactions to and opinions about "everything", is a new and different challenge. Some of the first results and recent analysis of Twitter sentiment data. Two main research areas of mining opinion operate either on the document level [13]. Both classification methods at the document level

and at the level of the sentence are generally based on the identification of opinion words or phrases.

The aim of this work is to implement our algorithm of semantic analysis that can advance our research. Our contribution propose a new algorithm for computing the semantic similarity between training tweets and the new test tweets using WordNet as an external network semantic resource.

3 Research Background

The needs for this project can be divided into four parts: the requirements related to the construction of an associated tweets training data the theme of feelings depressed, retrieving the tweets from Twitter, and needs related to the automatic classification using machine learning algorithms and the requirements for a semantic analysis of these sentiments to improve our results (Fig. 1).

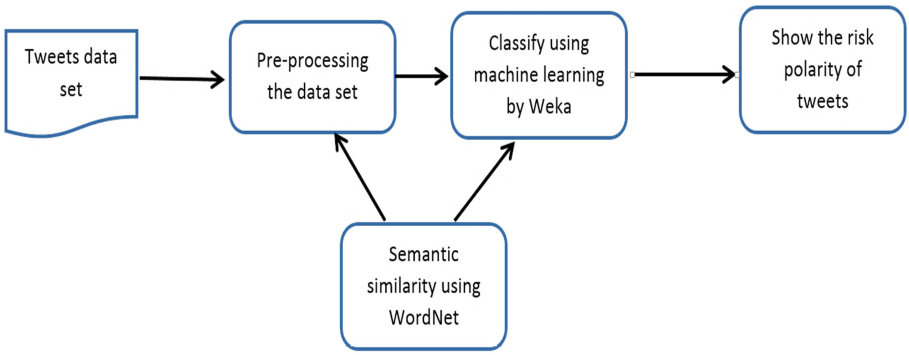


Fig. 1. Block diagram detailing our research work

3.1 Vocabulary

Before embarking our work, we bring to define a vocabulary associated with different themes of depression, fear, harassment, etc. Because most messages are published in English on Twitter, it was preferable to define the vocabulary in that language. Moreover, the vocabulary should be divided into different categories and sub-categories so you can easily identify the degree of threat from the tweet. For example, in the case of harassment via a tweet, it is necessary to identify the recipients of tweets (for it is they who may take action). In contrast, for other categories of vocabulary, it is the people who posted the tweet, which must be identified.

To store the vocabulary, our process must include a database, which will also be used to store suspicious tweets and the classification results (Fig. 2).

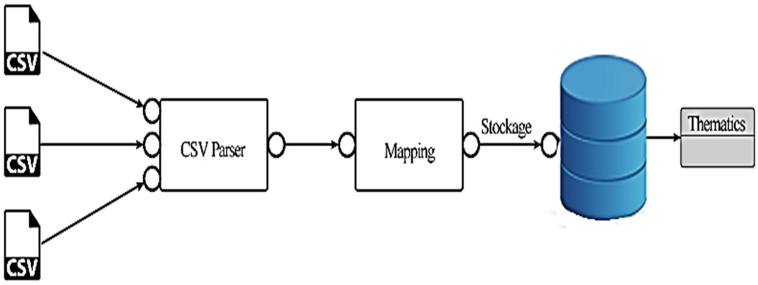


Fig. 2. Recovery and storage of our vocabulary

3.2 Retrieving Tweets

The collection of tweets is a second need in our project. The work to be developed should allow to automatically identify suspicious tweets (from the vocabulary defined previously) and stored for further analysis in the application's database.

In order to retrieve tweets from the Twitter API Rest of the following treatments must be performed (Fig. 3):

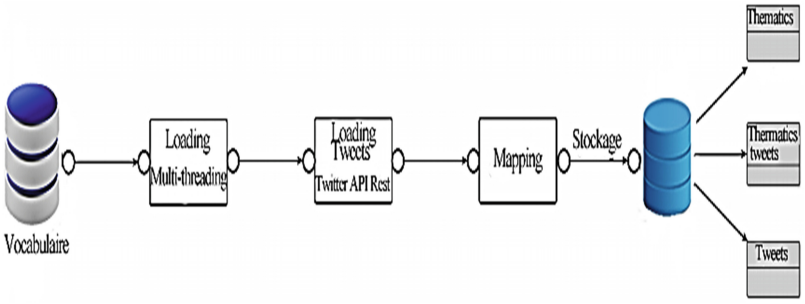


Fig. 3. Treatments to classify the Twitter data

3.3 Automatic Classification Tool

The integration of an automatic classification tool in the application enables a thorough analysis and automated "suspect tweets" stored in the database. We have identified two classes:

- Suspected tweets for which there is a high risk that the authors have a feelings of depression.
- Tweets suspects with a low risk.

These tweets "classified manually" have been loaded into the application and Weka uses them to learn a predictive model. Subsequently, Weka will be able to classify suspects tweets (stored in the "tweets" table) in both classes. As output, the application thus provides classified suspects tweets (Fig. 4).

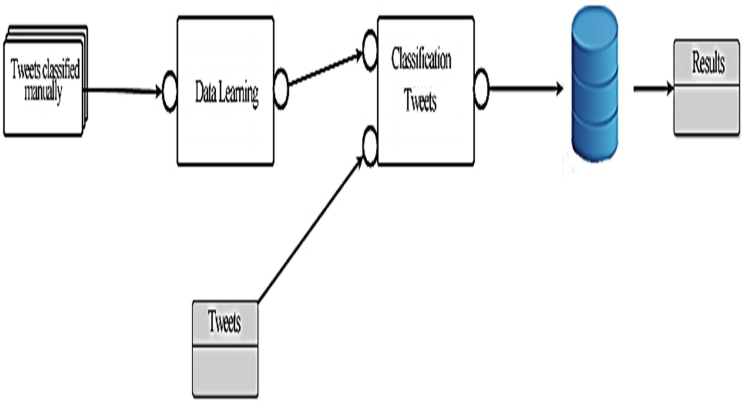


Fig. 4. The process of our method to compute the semantic similarity between the suspected tweet and the new tweets.

3.4 Semantic Analysis

Sentiment analysis is an method of analysis opinions, emotions and the feelings of people export to entities such as products, services, organized tions, events and subjects. Lexical processing of a lexicon notice to make the sentiment classification with a simple algorithm that assigns either a positive or a negative integer each tweet by counting the number of positive or negative words in the tweet. Before we are applying sentiment classification algorithm, we remove stop words, punctuation and numbers of data sets. We apply the sentiment classification at tweet to further divide the tweets sets extracted with keywords in two subsets of each set tweet. Each tweet is assigned zero, negative, or positive integer value after feeling algorithm is applied.

The aim of this work is to implement our method of semantic analysis that can advance our research. Our contribution propose a new method for computing the semantic similarity between suspected tweets and the new retrieval tweets using WordNet as an external network semantic resource.

A Proposed Algorithm

Our contribution propose an algorithm for computing the similarity between text documents and the query based on Wu and Palmer measure [14]. The principle of this measurement is given an ontology formed by a set of nodes and a root node (R). X and Y represent two ontology elements for which we will compute the similarity. The principle of similarity measurement is based on the distances (N1 and N2) which separate the X and Y nodes from the node R and the distance (N) which separates the Subsuming Concept (SC). The Wu and Palmer measurement is defined by this formula:

$$\mathrm{Sim}(\mathrm{X},\mathrm{Y})_{\text{Wu and Palmer}} = \frac{2 \times \mathrm{N}}{\mathrm{N1} + \mathrm{N2}} \tag{1}$$

To compute the semantic similarity between tweets data set and the tweets training set we apply the following algorithm:

- ***Input***: *TweetsData, training_tweets*
- *RemoveStopWord (TweetsData)*
- *RemovePunctuation (TweetsData)*
    - *For each element ∈ (TweetsData, termOftweet)*
    - *Return (TweetsData, term_tweet)*
    - *End for*
- *List(Q1) = indexing (training_tweets)*
- *S=0*
- *X ← 0*
- *Y ← 0*
    - *For each n ∈ List(term_tweet)*
    - *F=calculateoccurence(n)*
        - *For each e ∈ List(Q1)*
        - *R= calculateoccurence(e)*
        - *X←X+F×R×Sim(n, e)*
        - *Y←Y+F×R*
        - *End for*
    - *End for*
- *S←X/Y*
- *Return (tweet, S)*

Our similarity measure is presented by the following formula:

$$\mathrm{Sim}(q,d) = \frac{\sum_{i=1}^{n}\sum_{j=1}^{m} Q_i \times D_j \times \mathrm{SIM}(i,j)}{\sum_{i=1}^{n}\sum_{j=1}^{m} Q_i \times D_j} \tag{2}$$

i	: represents the terms of the query Q
j	: represents the terms of the new tweet D
q_i	: is the frequency of the term i in query Q
d_j	: is the frequency of the term j in new tweet D
$SIM(i,j)$	: is the similarity measure between the two terms i and j.

4 Experimental Analysis

For sentiment analysis related on the feeling of depression, we used the our algorithm to compute the similarity between the new tweets data set and the training data tweets set, because it requires us a very important measure based on Wu and Palmer approach and WordNet as an external network semantic resource.

The experiment was performed using the Weka tool. The choice of this tool is due to the fact that it is widely used in the field of machine learning and data mining, and it is also easy to handle. It is compatible with the data format we chose.

After loading the data into the tool, applying different functions for this study. Weka provides four options for evaluating the performance of the model learned in this experiment, we will consider only two of them. First, apply the algorithms on the entire data. The figure below shows the testing quality of the sentiment analysis of machine learning algorithm. Using Twitter API, tweets related to products are collected. A dataset is created using 200 twitter posts of electronic products.

Table 1. Cross-validation of performance on different classifiers

Algorithm	IBK	SMO	J48	CART
Precision	80,6%	78%	86,50%	87,6%
Recall	85,4%	84,9%	75,8%	83%
F-measure	82,9%	81%	80,08%	83,5%

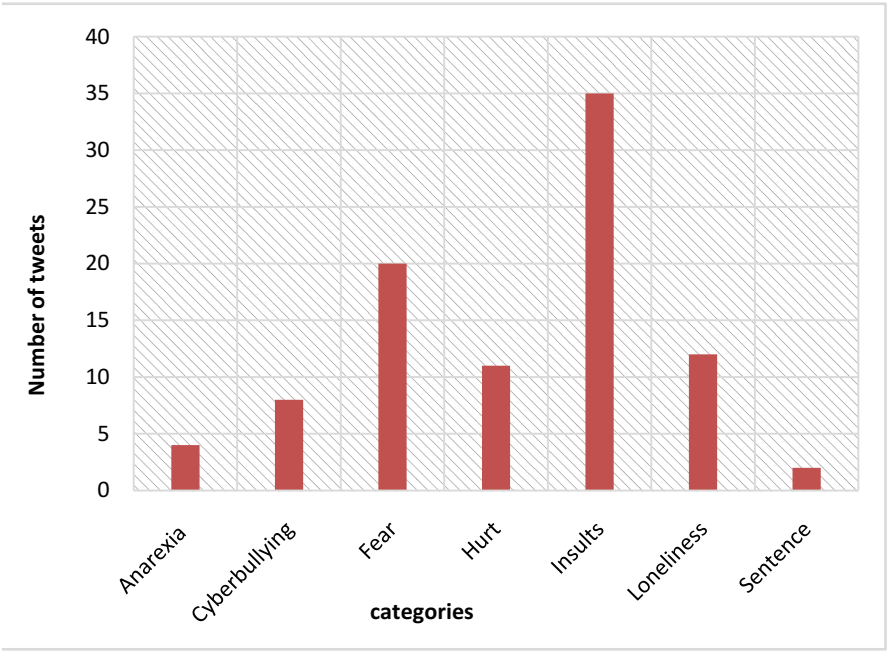


Fig. 5. Number of tweets by percentage of recovered categories

Figure 5 present the number of tweets by category (in percentage). The following tables show the various classification algorithms and the results we have achieved in terms of accuracy (Table 1):

Table 2 present the correctly classified instances and incorrectly classified instances are represented by classifier with cross-validation = 20.

Table 2. Visualization of cross-validation results

Classifier	Correctly classified instances	Incorrectly classified instances	Mean absolute error
J48	78.4314%	21.5686%	0.2449
IBK	74.5098%	25.4902%	0.2022
SMO	86.2745%	13.7255%	0.1373
PART	21.5686%	21.5686%	0.2677

5 Conclusion and Future Work

As part of this work, we present our method based on machine learning by using Weka for using the social network Twitter as a preventive force for estimating his/her depressive tendency. In addition, our work can analyze semantically the Twitter data based on WordNet. In our future work, we plan to further improve and refine our techniques in order to enhance the accuracy of our method. Thereafter, we involved to test multilingual WordNet for tweets.

References

1. Po-Wei Liang, B.-R.D.: Opinion mining on social media data. In: IEEE 14th International Conference on Mobile Data Management, pp. 91–96 (2013)
2. Statistic Brain. Twitter Statistics (2014). http://www.statisticbrain.com/twitter-statistics/
3. Li, R., Lei, K.H., Khadiwala, Chang, R.: TEDAS: a twitter-based event detection and analysis system. In: IEEE 28th International Conference on Data Engineering, ICDE 2012, pp. 1273–1276 (2012)
4. Chih-Hua T., Zheng-Han T., Yung-Sheng L., Yue-Shan C.: Mental disorder detection and measurement using latent Dirichlet allocation and SentiWordNet. In: IEEE International Conference on Systems, Man, and Cybernetics, pp. 1215–1220 (2015)
5. https://twitter.com/Freddy_E
6. https://twitter.com/CapitalSTEEZ_
7. Zhao, D., Rosson, M.B.: (n.d). http://research.ihost.com/cscw08-socialnetworkinginorgs/papers/zhao_cscw08_workshop.pdf
8. Ramirez-Esparza, N., Chung, Ewa Kacewicz, C.K., Pennebaker, J.W.: The psychology of word use in depression forums in English and in Spanish: testing two text analytic approaches. Association for the Advancement of Artificial Intelligence. www.aaai.org
9. Vinodhini, G., Chandrasekaran, R.: Sentiment analysis and opinion mining: a survey. Int. J. **2**(6) (2012)
10. Turney, P.: Thumbs up or thumbs down Semantic orientation applied to unsupervised classification of reviews. In: Proceedings of the Association for Computational Linguistics
11. Hu, M., Liu, B.: Mining and summarizing customer reviews. In: Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD 2004, pp. 168–177
12. Wilson, T., Wiebe, J., Hoffmann P.: Recognizing contextual polarity in phrase-level sentiment analysis. In: The Advanced Research and Development Activity (ARDA)
13. Wu, Y., Zhang, Q., Huang, X., Wu, L.: Structural opinion mining for graph-based sentiment representation. In: Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing, EMNLP-2011 (2011)
14. Wu, Z., Palmer, M.: Verb semantics and lexical selection. In: Proceedings of the 32nd Annual meeting of the Associations for Computational Linguistics, pp 133–138 (1994)

Solving the Traveling Salesman Problem Using Ant Colony Metaheuristic, A Review

Sonia Kefi[1(✉)], Nizar Rokbani[1,2], and Adel M. Alimi[1]

[1] REGIM-Lab: Research Groups in Intelligent Machines, University of Sfax, ENIS, Sfax, Tunisia
{sonia.kefi, nizar.rokbani, adel.alimi}@ieee.org
[2] High Institute of Applied Sciences and Technology of Sousse, University of Sousse, Sousse, Tunisia

Abstract. This paper presents a software application allowing to solve and compare the key metaheuristic approaches for solving the Traveling Salesman Problem (TSP). The focus is based on Ant Colony Optimization (ACO) and its major hybridization schema. In this work, the hybridization ACO algorithm with local search approach and the impact of parameters while solving TSP are investigated. The paper presents results of an empirical study of the solution quality over computation time for Ant System (AS), Elitist Ant System (EAS), Best-Worst Ant System (BWAS), MAX–MIN Ant System (MMAS) and Ant Colony System (ACS), five well-known ACO algorithms. In addition, this paper describes ACO approach combined with local search approach as 2-Opt and 3-Opt algorithms to obtain the best solution compared to ACO without local search with fixed parameters setting. The simulation experiments results show that ACO hybridized with the local search algorithm is effective for solving TSP and for avoiding the premature stagnation phenomenon of standard ACO.

Keywords: Hybrid metaheuristic · Ant Colony Optimization · AS · EAS · BWAS · MMAS · ACS · Traveling Salesman Problem · 2-Opt algorithm · 3-Opt algorithm

1 Introduction

Combinatorial optimization occupies a very important place in operations research, discrete mathematics and computer science. Firstly, its importance is justified by the great difficulty of optimization problems and secondly by many practical applications which can be formulated as a combinatorial optimization problem. These problems are often easy to define but they are generally difficult to solve. Indeed, most of these problems belong to the class NP-hard problems, but until now, it has not an effective solution for all information.

Several problems are considered as combinatorial optimization problems, including the routing problems as one of the most popular and difficult to solve. The most known problems and the most used of routing problems are the Traveling Salesman Problem (TSP) [1] and the Vehicle Routing Problem (VRP) [2], which are the subject of several literature researches [3].

A. Abraham et al. (eds.), *Proceedings of the 16th International Conference on Hybrid Intelligent Systems (HIS 2016)*, Advances in Intelligent Systems and Computing 552,
DOI 10.1007/978-3-319-52941-7_42

The Traveling Salesman Problem (TSP) is the problem of finding the shortest path connecting n cities. The Salesman must visited each city only once and return to the starting city. The difficulty of the problem is the combinatorial explosion of solutions to explore when the number of cities visited is increased. This problem is considered as a NP-complete problem.

More formally, TSP problem can be modeled as a complete undirected graph G and a function distance d. TSP is a discrete optimization problem. Several methods specifically has been developed to solve TSP and mathematically and experimentally investigated. However, the TSP is also well suited to be approached by *metaheuristic* methods that provide close optimal solutions with a reasonable execution time [4].

Metaheuristic methods belong to the class of approximate problem-solving methods that try to combine various heuristic approaches to explore efficiently the solution space of a problem. These methods are based on solution of the problem as neighborhood search methods like Taboo Search [5], Swarm Intelligence methods like Particle Swarm Optimization (PSO) [6] and Ant Colony Optimization (ACO) [7], etc.

In the paper [8], the authors describes an improved ACO method for solving TSP. They have modified the strategy of pheromone updated and have changed parameters setting. Finally, a local optimal search strategy is introduced to avoid the premature stagnation phenomenon in the convergence process. In [9–11], the authors presents hybrid method based on ACO and PSO named AS-PSO to solve TSP and this method has shown its efficiency compared to single ACO, in terms of time and solution. In [12] and [13], the authors used a hybrid method based on ACO, PSO and local search approach as 2-Opt algorithm. This method improves local solution and finds better path solutions for key TSP test benches than ACO.

This paper describes a hybrid metaheuristic based on Ant Colony Optimization (ACO) and Local Search approach to solve TSP. This combination proves the performance of the approach and it advantage is the capacity to achieve the best global solution than ACO without local search. Several extensions of ACO algorithm are implemented and tested using TSPLIB benchmark [14].

The ACO algorithm is combined with 2-Opt and 3-Opt algorithms to improve the path tour. The construction of path based-method as ACO algorithm can achieve acceptable solutions, but with high computational time. For the improvement of path based-technique like 2-Opt and 3-Opt algorithms produced quickly results but it cannot realize a good solution in an acceptable time. Finally, in this hybrid metaheuristic method, we combine the two types of algorithm to find good solutions than other methods studies in the literature, used same materials and algorithms.

This paper is organized as follows: In Sect. 2, we present an overview of ACO for solving TSP. In Sect. 3, we describe the hybrid metaheuristic approach based on ACO and Local Search methods. In Sect. 4, we realize some experimentations. Finally, in Sect. 5, we summarize the results and we emphasize on the relevance of ACO.

2 Ant Colony Algorithm Overview for TSP

The metaheuristic Ant Colony optimization (ACO) is relatively recent. It is very popular and has been applied successfully to combinatorial optimization problems as TSP since 1994.

2.1 Basic Principles of Ant System Algorithm

The first ACO algorithm is Ant System (AS) [7]. It was introduced using the TSP as an example application. At any time *t*, each ant chooses a destination city within a defined range. All ants are placed at time *(t + 1)* in a city of their choice. After n iterations, all of the colony construct a Hamiltonian circuit on the graph.

It is specified that each ant has a memory implemented by a list of cities already visited. This ensures that no ant will visit twice the same city at the heart of its research. The memory of each ant is emptied when they have completed their cycle.

Each ant *k* is placed on the city *i* at time *t* will choose its destination city *j* according to the η_{ij} visibility of this city and the amount of pheromones τ_{ij}(t) deposited on the arc connecting the two cities. This choice will be made randomly, with a probability of selecting the city *j* given by the Eq. 1:

$$p_{ij}^{k}(t) = \begin{cases} \frac{\left[\tau_{ij}(t)\right]^{\alpha}\cdot\left[\eta_{ij}\right]^{\beta}}{\sum_{k\in\Omega i}\left[\tau_{ik}(t)\right]^{\alpha}\cdot\left[\eta_{ik}\right]^{\beta}} & \text{if j} \in \Omega\text{i} \\ 0 & \text{Otherwise.} \end{cases} \quad (1)$$

Where Ωi presents all cities that ant *k*, placed on the city *i*, has not yet visited at time *t* in the current cycle. α and β are two parameters that control the relative importance between pheromones and visibility. Therefore, if α is equal to 0 and β is different from 0, the choice will be made only in terms of visibility.

$\eta_{ij} = \frac{1}{d_{ij}}$ presents the visibility of a city *j* when one is placed on the city *i*.

At the end of each cycle (each ant has reached the n vertices that make up the graph), the variables pheromones are updated according to the Eq. 2:

$$\tau_{ij}(t+1) = (1-\rho).\tau_{ij} + \sum_{k=1}^{m}\Delta\tau_{ij}^{k}. \quad (2)$$

Where $\rho \in]0, 1]$ defines the rate of pheromones evaporation on the arcs between time *t* and time *(t + 1),* and $\Delta\tau_{ij}$ (t) is the amount of pheromone deposited by ants in the same time slot on the arc (i, j).

$\Delta\tau_{ij}^{k}$(t) represents the amount of pheromone deposited by the k^{th} ant on arc *(i, j)* in the same time interval, so, it is defined by the Eq. 3:

$$\Delta\tau_{ij}^{k} = \begin{cases} \frac{1}{L_k(t)}, & \textit{if } arc(i,j) \in T^k \\ 0 & \textit{Otherwise} \end{cases} \quad (3)$$

Where L_k, the length of the tour T_k built by the k^{th} ant, is computed as the sum of the lengths of the arcs belonging to T_k.

2.2 Key Extensions of Ant System Algorithm

Several research try to expand and to improve the AS performance by controlling better the intensification and diversification of the search process. Several extensions are developed and they improved performance of AS. In fact, most of these extensions keep the same solution construction procedure of AS algorithm as well as the same pheromone evaporation procedure. These extensions include Elitist AS, Best-Worst AS, MAX–MIN AS and Ant Colony System. The main differences between AS and these extensions are the pheromone update which is performed, as well as some additional details in the pheromone trails.

2.2.1 Elitist Ant System

A first improvement on the initial AS, called the Elitist strategy for Ant System (EAS), it was introduced by Dorigo [7]. The idea is to provide strong additional reinforcement to the arcs belonging to the best-so-far tour (T_{bs}) found since the start of the algorithm. Note that this additional feedback to the best-so-far tour which can be viewed as additional pheromone deposited by an additional ant called best-so-far ant, which is another example of a daemon action of the ACO metaheuristic.

2.2.2 MAX–MIN Ant System

Stützle and Hoos introduced the MMAS algorithm [15] to improve ACO performance. The MMAS combines an improved exploitation of the best solutions found during the search mechanism with an efficient to prevent premature stagnation research. MMAS algorithm introduces three main modifications compared to AS:

- After found the best tour, only the iteration-best ant, that is, the ant that produced the best tour in the current iteration, or the best-so-far ant is allowed to deposit pheromone.
- To avoid stagnation of research, the possible values of the amount of pheromone on each solution component is limited to the interval $[\tau_{\min}, \tau_{\max}]$.
- Pheromone trails are initialized to the upper pheromone trail limit $\tau_{\max}$ to ensure high exploration of the solution space at the beginning of the algorithm.

2.2.3 Ant Colony System

The Ant Colony System (ACS) [16, 17] algorithm was introduced by Dorigo and Gambardella to improve the performance of large scale problems.

ACS introduced a rule that depends on a parameter q_0 ($0 <= q_0 <=1$), which defines a balance diversification/intensification. This rule allows ants to promote more the arc which contains the following maximum trace q_0, otherwise the usual transition rule is used. Thus, ACS allows intensify more research towards the most promising areas, i.e., containing more than traces.

3 Ant Colony Algorithm with Local Search

From the literature, metaheuristic approach can give a best optimal solution when it is combined with a local search algorithm [7, 16]. To solve the TSP, the metaheuristic approaches apply local search to the initial solution to obtain a local optimal solution.

The architecture of ACO to solve TSP includes the possibility of using local search (see Fig. 1). When, ants have completed their tour, the optimal solution can be improved by a local search as k-opt. Then, pheromones are updated on the arcs of the local optimized solution. The combination of ACO algorithm with local search algorithm is called as hybrid approach. In the next section, we will study the performance of different ACO extensions algorithms presented before and we will demonstate that AS is improved when it is coupled with a local search method. Therefore, we have implemented two of the most used types of local search for the TSP: 2-Opt and 3-Opt algorithms.

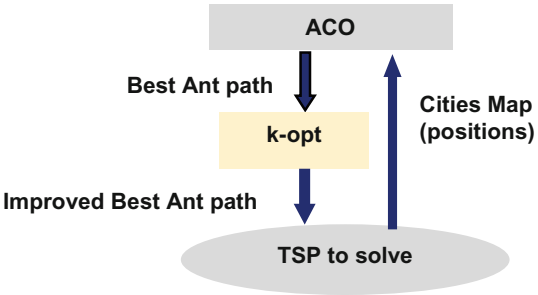


Fig. 1. Hybrid AS with local search architectural overview.

4 A Software Application to Solve TSP Based on ACO Variants

To conduct the experimental results related to our paper, a software application was developed used C++ language. Figure 2 presents the first interface when we open the application. Therefore, our application allows:

- To select the ACO variants as AS, ACS, Elitist AS, Best-Worst AS, Min-Max AS.
- To choose TSP file which contains the test bench to be executed. When we select TSP test bench, the optimal course is traced in the interface as in Fig. 2 which presents optimal course of berlin52 problem and optimal distance of this course is shown.
- To select the heuristic information. α and β are selected in [1, 5] and the rate of pheromones evaporation ρ is selected in]0, 1].
- The parameters of ACS are enabled only when the user select the ACS algorithm to execute. Therefore, the parameters q_0, τ_0 and ξ of local pheromone update rule are selected in [0.1, 1].

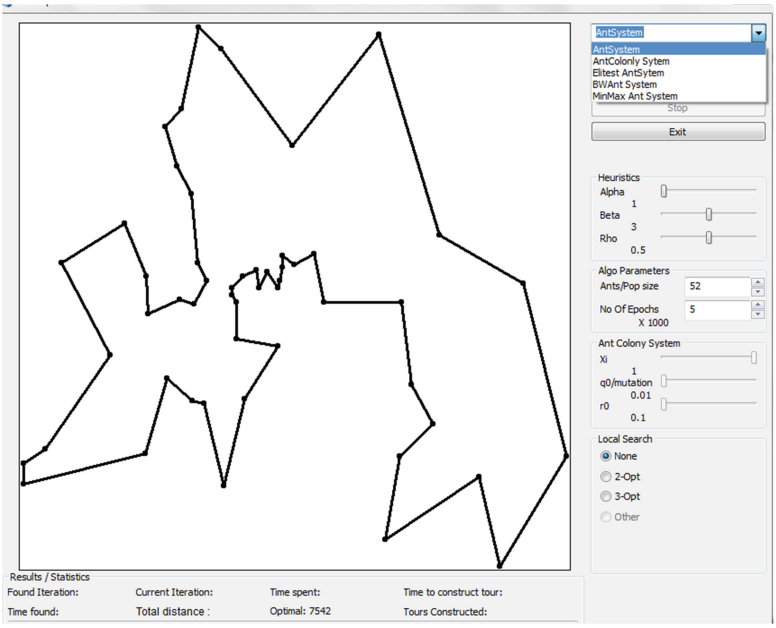


Fig. 2. Initial interface of our software application.

- To fixe freely the algorithm parameters as number of ants and number of iteration.
- To choose the local search algorithm. The user can select to executed ACO without local search (None), with 2-Opt or 3-Opt.
- Finally, we can obtain some results and statistics after execution of our application. We can show the total distance of the path obtained, the minimum number of cycles and the time found in second (s) to achieve best solution (Found Iteration in Fig. 2), the time spent and the time allowed to construct a tour (in s).

4.1 Hardware/Software Platform

Hardware/software environment used in the experiment are Core i5-2430 M CPU 2.40 GHz, 4 GB memory, 500 GB for hard disk capacity and Microsoft Windows 7 (Service Pack 1). The development platform was the Microsoft Visual C++.

4.2 ACO Parameters Setting

A major weakness of AS is the high number of parameters involved. Therefore, it is very difficult to select them subtly. In the absence of rigorous argument, the only justifications available to researchers for fixing AS parameters are experimental results. The experiments were carried out on sets of cities. Some conclusions were learned and we present them in some points:

1. We set $\rho = 0.5$, only the greedy strategy is to work at the beginning of the algorithm.
2. The search is more effective when the amounts of pheromones have a great impact on the policy decision of the ant. This corresponds to take α is equal to 1. In addition, for the parameter β which represents the importance of heuristic factor, it is in between 2 and 5, so we have used the $\beta = 3.4$.
3. There is an optimum when the number of cities m is equal to number of ants n.

Table 1 summarizes the choice of ACO parameters setting given by Dorigo in [7] and used in all TSP test benches.

Table 1. ACO Parameters setting used in all experimental.

Parameters	Values
Number of ants	Number of cities
Maximum iteration	5000
α	1
β	3.4
ρ	0.5
ACS transition rule parameters	$q_0 = 0.9$ $\xi = 0.1$

4.3 Simulation and Discussions

To evaluate the proposed method, a statistical analysis was performed through 5000 executions using key test benches form TSPLIB [15]. The comparison of hybrid method with other research from literature are given in Table 2. We can conclude that the hybrid AS-2Opt and AS-3Opt method give interesting results compared to standard AS and the best known solutions, BKS of all test benches described before. These solutions are presented in bold when hybrid AS method performs better. Also, Table 2 shows that the time found of our hybrid AS is very short and better compared to the time found by standard AS. Therefore, we can conclude that hybrid AS with local search approach is an effectiveness method and give best result compared standard AS.

Moreover, we can detect that for the problem eil51, berlin52 and eil76 hybrid AS method gives the same result of the best known solution, BKS as 426, 7542 and 538 and for standard AS, results are close to optimal solution. Finally, through the obtained results described in Table 2, we can synthesize that the AS-2Opt and the AS-3Opt methods achieve judicious for all test benches.

In Table 3, we have applied other variant of ACO as ACS, MMAS, Elitist AS and BWAS for all test benches when AS with local search do not perform better as in st70 test. The best solutions are presented on bold. Finally, we can conclude that for each TSP test bench we can apply a specific variant of ACO that gives the best solution for the test. The results shows that ACO with local search method performs better than standard ACO in term of result obtained and computational time.

Table 2. Comparison of experimental results between standard AS and hybrid AS.

Problem	BKS	Local Search	Best Solution	Nb of cycles to achieve best solution	Time Found (s)
eil51	426	None	435	4458	10.68
		2-Opt	428	4211	10.39
		3-Opt	**426**	1276	4.13
berlin52	7542	None	**7542**	4720	12.42
		2-Opt	**7542**	144	0.42
		3-Opt	**7542**	62	0.24
eil76	538	None	546	3424	25.4
		2-Opt	544	1140	8.6
		3-Opt	**538**	829	13.73
st70	675	None	702	852	9.96
		2-Opt	700	864	8.4
		3-Opt	**696**	2508	17.75
ch150	6528	None	6701	2087	117.91
		2-Opt	6670	332	15,97
		3-Opt	**6631**	820	45.83
kroA200	29368	None	32135	1906	246.04
		2-Opt	32046	1538	189.85
		3-Opt	**31418**	1854	231.7

Table 3. Comparison of experimental results between some variant ACO and hybrid ACO with local search.

Problem	BKS	Algorithm	Local Search	Best Solution	Min Nb of cycles to achieve best solution	Time Found (s)
st70	675	ACS	None	676	414	6.01
			2-Opt	**675**	155	2.42
			3-Opt	**675**	932	13.61
		MMAS	None	682	2598	15.02
			2-Opt	**675**	286	1.5
			3-Opt	**675**	725	4.3
		Elitist AS	None	691	4493	26.22
			2-Opt	**678**	2344	14.27
			3-Opt	689	2139	12.98
		BWAS	None	687	2109	11.59
			2-Opt	**684**	934	5.16
			3-Opt	**684**	1464	7.71
ch150	6528	ACS	None	6556	902	73.64
			2-Opt	**6549**	385	31.65
			3-Opt	**6549**	1077	89.20
		MMAS	None	6614	238	10.01

(*continued*)

Table 3. (*continued*)

Problem	BKS	Algorithm	Local Search	Best Solution	Min Nb of cycles to achieve best solution	Time Found (s)
			2-Opt	**6547**	1076	42.54
			3-Opt	6544	1252	49.95
		Elitist AS	None	6684	1744	83.64
			2-Opt	6645	1794	88.84
			3-Opt	**6594**	1868	94.04
		BWAS	None	6574	1080	47.96
			2-Opt	6556	1403	63.95
			3-Opt	**6549**	606	26.74

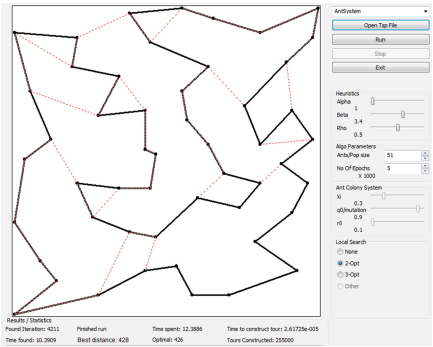


Fig. 3. Optimum tour of AS with 2Opt for eil51 test bench.

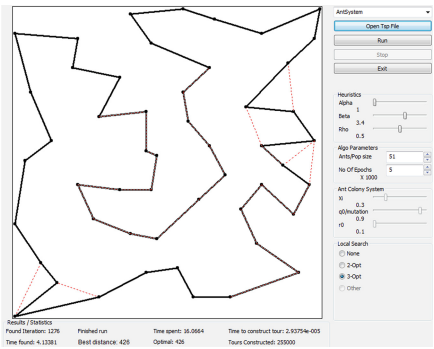


Fig. 4. Optimum tour of AS with 3Opt for eil51 test bench.

Figures 3 and 4 show the optimum course of the eil51 test bench, using respectively ACO with 2Opt and ACO with 3Opt. The best result is given by ACO with 3Opt, which is equal to 426.

5 Conclusion and Perspectives

This paper presents the ACO metaheuristic combined with local search method as 2Opt and 3Opt algorithm. This metaheuristic method has proven its performance for the TSP problem. The use of the local search decrease the probability of falling in local minimum. The simulation results verify the effectiveness of our algorithm.

Future improvement concern essentially the premature stagnation phenomenon of standard ACO so a perturbation method is needed. This platform will be made available to interested researchers its extension to major heuristics is ongoing.

Acknowledgments. The authors would like to acknowledge the financial support of this work by grants from General Direction of Scientific Research (DGRST), Tunisia, under the ARUB program.

References

1. Laporte, G.: The Traveling Salesman Problem – an overview of exact and approximate algorithms. Eur. J. Oper. Res. **59**, 231–247 (1992)
2. Christofides, N.: The vehicle routing problem. RAIRO – Oper. Res. **10**, 55–70 (1976)
3. Wang, C., Mu, D., Zhao, F., Sutherland, J.W.: A parallel simulated annealing method for the vehicle routing problem with simultaneous pickup delivery and time windows. Comput. Ind. Eng. **83**, 111–122 (2015)
4. Alba, E.: Parallel metaheuristics: a new class of algorithms. In: Wiley-Interscience (2005)
5. Glover, F.: Tabu search. ORSA J. Comput. **1**(3), 190–206 (1989)
6. Kennedy, J., Eberhart, R.: Particle swarm optimization. In: IEEE International Conference on Neural Networks, vol. 4, pp. 1942–1948 (1995)
7. Dorigo, M., Stutzle, T.: Ant Colony Optimization. Bradford Company, Scituate (2004)
8. Yang, X., Wang, J.S.: Application of improved ant colony optimization algorithm on traveling salesman problem. In: Chinese Control and Decision Conference, pp. 2156–2160 (2016)
9. Rokbani, N., Momasso, A.L., Alimi, A.M.: AS-PSO, Ant supervised by PSO meta-heuristic with application to TSP. In: Proceedings Engineering & Technology, vol. 4, pp. 148–152 (2013)
10. Rokbani, N., Abraham, A., Alimi, A M.: Fuzzy ant supervised by PSO and simplified ant supervised PSO applied to TSP. In: The 13th International Conference on Hybrid Intelligent Systems (HIS), pp. 251–255 (2013)
11. Kefi S., Rokbani N., Krömer P., Alimi A.M.: A new ant supervised-PSO variant applied to traveling salesman problem. In: The 15th International Conference on Hybrid Intelligent Systems (HIS), pp. 87–101 (2015)
12. Kefi, S., Rokbani, N., Krömer, P., Alimi, A.M.: Ant supervised by PSO and 2-opt algorithm, AS-PSO-2Opt, applied to traveling salesman problem. In: IEEE International conference on System Man and Cybernetics SMC (2016)
13. Kefi, S., Rokbani, N., Alimi, M.A.: Hybrid metaheuristic optimization based on ACO and standard PSO applied to traveling salesman problem. Int. J. Comput. Sci. Inform. Secur. **14**(7), 802–823 (2016)
14. Reinelt, G.: TSPLIB-a traveling salesman problem library. ORSA J. Comput. **3**, 376–384 (1991)
15. Stützle, T., Hoos, H.: The MAX-MIN ant system and local search for the traveling salesman problem. I*n:* International Conference on Evolutionary Computing and Evolutionary Program, pp. 13–19 (1997)
16. Dorigo, M., Gambardella, L.M.: Ant colony system: a cooperative learning approach to the traveling salesman problem. IEEE Trans. Evol. Comput. **1**(1), 53–66 (1997)
17. Gambardella, L.M., Dorigo, M.: Ant Colony System hybridized with a new local search for the sequential ordering problem. Inform. J. Comput. **12**(3), 237–255 (2000)

Hybrid Neural Network and Genetic Algorithm for off-Lexicon Online Arabic Handwriting Recognition

Yahia Hamdi(✉), Aymen Chaabouni, Houcine Boubaker, and Adel M. Alimi

REGIM-Laboratory: REsearch Groups in Intelligent Machines, National School of Engineers, University of Sfax, BP 1173, 3038 Sfax, Tunisia
{yahia.hamdi.tn,ayman.chaabouni, houcine-boubaker,adel.alimi}@ieee.org

Abstract. In this paper we propose the hybridization of neural networks and genetic algorithm for online Arabic handwriting recognition. The used method consists in decomposing the input signal into continuous parts called graphemes based on Beta-Elliptical model and baseline detection. The segmented graphemes are then described according to their position in the pseudo-word by a combination of geometric features modeling their trajectory shape and provided in the input of the neural networks used for graphemes class recognition. Finally, a genetic algorithm is used to generate the characters code corresponding to the obtained chain of recognized graphemes code by applying the genetic search process: selection, crossover and mutation. The developed system is evaluated using an Arabic words dataset extracted from the ADAB Database.

Keywords: Hybrid recognition system · Off-lexicon recognition · Online Arabic handwriting · Neural networks · Genetic algorithm

1 Introduction

The handwriting recognition domain with his two branches offline and online has been of interest during the two last decades. Lately, Online Handwriting Recognition has become a popular area of research because of the advances in technology such as the handwriting capturing devices as tablet-PC, smart phones and Digital Assistants (PDAs). Nowadays, these devices are frequently used by most young people around the world, in increasingly off-lexicon textual manner. In fact, their scripts become more and more composed of mixed and expandable vocabulary by including abbreviations, foreign slogans and international denominations, handwritten with the common alphabet graphical code adopted by the writer. This fact makes the task of handwriting recognition more challenging especially when it deals with cursive Arabic language by imposing an unlimited lexicon context. The Arabic language already has a large properly lexicon. Several Researchers [1, 2], developed their approaches for small vocabulary while other works have introduced an extended vocabulary system for online Arabic handwriting recognition. Considering the challenges imposed by

A. Abraham et al. (eds.), *Proceedings of the 16th International Conference on Hybrid Intelligent Systems (HIS 2016)*, Advances in Intelligent Systems and Computing 552,
DOI 10.1007/978-3-319-52941-7_43

supporting small and large vocabulary lexicons for Arabic language, we propose an off-lexicon online Arabic handwriting recognition system based on hybrid of neural networks and genetic algorithm. The main contribution of our work consists in segmenting an Arabic word into small segment or grapheme based on the Beta-Elliptical modeling. These graphemes are classified according to their position in the word into five sub-groups namely BG (Beginning Graphemes); MG (Middle Graphemes); IG (Isolated Graphemes); EG (End graphemes) and DG (Diacritics graphemes). The statistics of graphemes classification as described in [3]. After that, the combination of the enhanced Beta-Elliptical strategy [4] and Fourier descriptor is used to extract the features vector from each grapheme. These vectors are subsequently used as input for an MLP neural network in classification stage to recognize the chain of grapheme code that constitutes the handwritten script. Finally, a genetic algorithm is used to deduct the chain of similar characters.

The paper is organized as follows: In Sect. 2, we describe the related work. Section 3 presents our method. The following section describes experiments and results using graphemes databases. Finally, the conclusion with some future works is reported in Sect. 5.

2 Related Work

In the last few years, intensive research has been done to solve the problem of Online/Offline handwriting recognition systems. Compared with offline recognition fields, research studies about the Online Arabic handwriting recognition are still very few. Many researchers have been attracted towards this field and several approaches have been made during the last decade. In [5], a method based on Freeman-like chain code was developed and writer independent features are extracted for the recognition of handwritten Arabic characters. Mezghani [6] presented an online system for the recognition of handwritten Arabic characters using a Kohonen network which runs by an unsupervised clustering algorithm. Another analytical approach involves decomposing the word into smaller units or primary and secondary strokes. Their advantage is to work on an extended or infinite lexicon. A neuro-fuzzy system, which is online writer dependent, was developed by [7] to recognize Arabic words. The last system uses a neuro-physiological description of the online script based on Beta impulses overlapping for the velocity profile modeling. Recently, Eraqi [1] presented a segmentation method for on-line Arabic handwriting based on the baseline detection. This method uses the projection of the writing to detect the levels of the baselines which is considered an important step for the determination of delay strokes. Also, Boubaker [8] developed a system for online Arabic handwriting trajectories modeling based on graphemes segmentation. It consists of baseline detection which is used as a reference to extract the particular points delimiting the graphemes segmentation. Likewise a hybrid method combining NN/HMM classifiers seems to be promising to take advantages of large and non-linear context modeling via neural networks while profiting from the Markovian sequence modeling [9]. An Arabic handwritten recognition system based on matching template algorithm and genetic algorithm was developed [10]. They used a heuristic method by using 13 visual codes to represent the

handwritten trajectory. A novel idea investigated two deep architectures: Deep Belief Network (DBN) and Convolutional Neural Networks (CNN) which are applied for recognizing Arabic handwritten script [11].

3 System Overview

Our contribution is mainly focused on hybrid approach for creating an off-lexicon online handwriting recognition method based on combination of neural networks and genetic algorithm. Our proposed method composed of five stages. It include the pre-processing step which aims to normalize the handwriting size in order to minimize the noise by applying a Chebyshev type low pass filter with a cut-off frequency of fcut = 12 Hz, on the acquired trajectory [12]. Second, the segmentation technique involves the baseline detection and the graphemes delimiting trajectories. After that, the combining beta-elliptic characteristics and trajectory shape modeling are used for

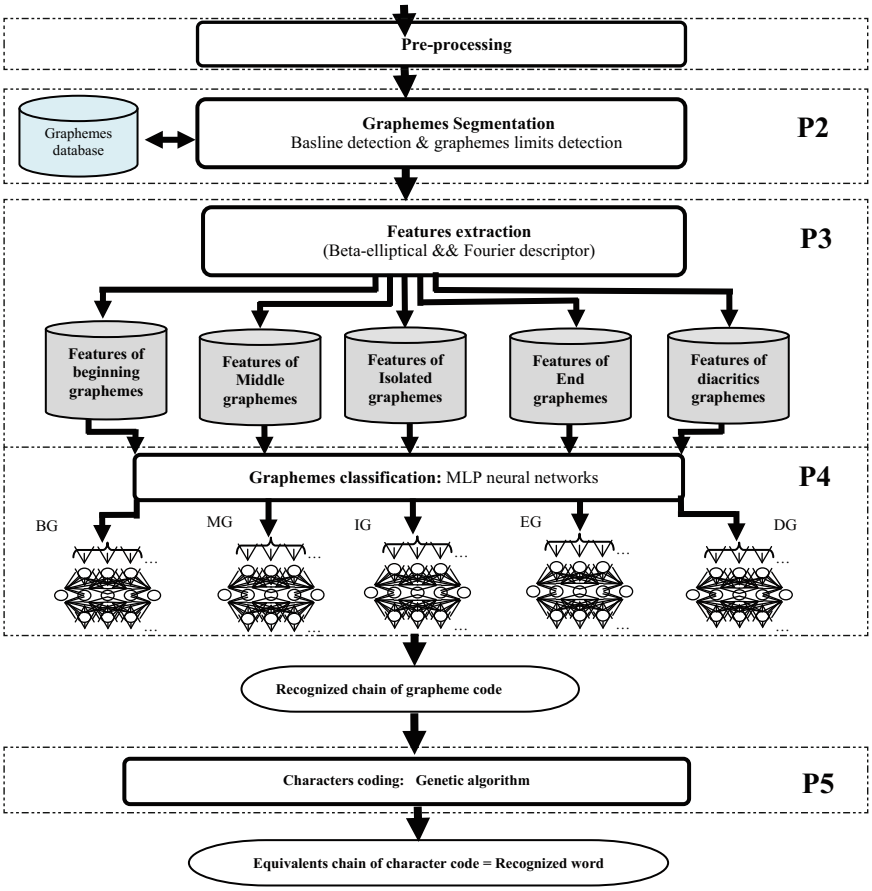


Fig. 1. Proposed method overview

features extraction. We use an MLP neural networks that takes as input the features vector for graphemes recognition. Finally, a genetic algorithm is used to generate the optimized characters code corresponding to the obtained chain of recognized graphemes code. The constructed graphemes database which is used for our experimentation is created from the Arabic words extracted from ADAB database. The Fig. 1 represents an overview of our architecture method.

3.1 Graphemes Segmentation

In the analytical approach of word recognition, segmentation is one of the most critical tasks. The aim of segmentation is to split the entire word into elementary writing components called graphemes that represents basic gestures and graphic shapes. For example, many Arabic characters such as 'س', 'ب' include several basic form called 'nabra' (ـ) graphemes. In our work, the adopted segmentation technique is based on the detection of the handwriting baseline direction using the algorithm proposed by [8]. The obtained local baseline direction permits then to extract the graphemes limits from the set of Beta-elliptic strokes endpoint that correspond to: the Bottom of the ligature valleys and Angular points [8]. Each segmented grapheme is then pre-classified according to its position in the word into five sub-classes corresponding to BG, MG, IG, EG and DG. The above described procedure is applied on the ADAB database of online cursive Arabic handwriting to create a new grapheme database used for grapheme shape learning and also integrated in the designed off-lexicon recognition process.

3.2 Handwriting Modeling

The purpose of feature extraction stage is to produce an appropriate set of features. For this we have used the combination of Beta-elliptic model and Fourier descriptor. The Beta-elliptic model is used in many field of research for online handwriting, such as in regeneration of handwriting [13] and in writer identification [14]. In the field of the online handwriting modeling, the Beta-elliptic model is characterized by a description combining dynamic and static profiles. In the dynamic profile, the velocity is modeled by overlapped Beta signals, while in the static profile called also geometric profile; segmented trajectory is modeled by elliptic arcs as explained in [4]. In order to discriminate the various types of graphemes, we extract two features' vector for each sub-groups of segments as shown P3 of Fig. 1. The resulting of the first feature vector is composed of 22 parameters describing the shape and position of the trajectory graphemes relative to the baseline. In addition of dynamic and static feature that representing handwriting trajectory from beta-elliptic model; we extract for each segmented grapheme a set of parametric features that represent dimensions, shape, and position from the handwriting script using a Fourier descriptors modeling. The second type of vector composed of 46 parameters correspond to combine the geometric parameters of localization with the Fourier descriptors for grapheme shape modeling and the features representing the extreme of the vertical and horizontal projection histograms of the fuzzy template.

3.3 Graphemes Classification

Based on the extracted features, the next phase attempts to identify the grapheme that represents the input features. We used a feed forward multilayer perceptron neural network for classification phase. Since the discriminating power of a neural network is related to the nature of the information data used for its training [15]. Given the context of our application of handwriting recognition, our method consists in classifying the graphemes into five groups (see P4 of Fig. 1). Thus, we devote an MLP neural network for each vector type. Three layers were used in this present research: the input layer, the hidden layer and the output layer. The network parameters used for training are: (Learning rate coefficient = 0.05, number of unit in input layer = 22 or 46 depending on the type of vector, number of unit in hidden layer = 50, transfer function used for hidden layer = 'Tansig', max epoch = 1000, number of unit in output layer: is the number of graphemes classes in the correspondent position).

Table 1. Different type of 'nabra' class

Categories of 'nabra' graphemes	Classes		
	Beginning	Middle	End
Restreinted nabra	ـد	ـدـ	ـا
Nabra with elongation in the right		ـدـ	ـا
Nabra with elongation in the left	ـد	ـدـ	

The segmentation of the input word into sub-units or graphemes is among the most challenging for the recognition of cursive script. One of these problems confronted are related to the presence in the Arabic script of 'nabra' which can be classified into three categories (see Table 1). Thus, used MLP neural networks; a post-classification step is applied to distinguish the different types of 'nabra' graphemes.

3.4 Characters Coding

After graphemes classification stage which aims to classify each grapheme of the handwriting trajectory in it's according most probable class. The next step consists in reorganizing and adapting the obtained chain by finding the nearest stable code graphemes chain able to be translated immediately in corresponding code characters chain. Four grapheme characteristics are considered in this task:

- The grapheme stability coefficient (C_{GS}) which represents the ability of the current grapheme to constitute a whole character alone. For example, the scripts 'و', 'ر' represents both a graphemes and whole characters.
- The grapheme neighboring stability coefficient (C_{GNS}) which represents the ability of the current grapheme to merged with other graphemes in the neighboring to constitute a whole character. For example, the character 'سـ' is formed by the association of three graphemes: beginning nabra 'ـد', Middle nabra 'ـدـ' and middle nabra with elongation in the left 'ـدـ'.

- The grapheme accuracy coefficient (C_{GA}) which represents the probability of its class recognition.
- The coefficient of grapheme differentiation (C_{GD}) which represents the cost of penalization attributed to the change of the class of the current grapheme by another grapheme class in proportion to the difference between their shapes.

Based on these grapheme characteristics, the algorithm seeks the best combination of graphemes code to build the adequate character code using the genetic algorithm.

The Genetic Algorithms (GA) are particularly well suited to the research of optimization problems where solutions to optimize the function may not be differentiable and has several local minima. In our case, genetic processing allows generating virtual character codes by applying the crossover and mutation to the grapheme codes of the decomposed words. Choosing the best character codes is established by a fitness function that will be described later.

3.4.1 Coding of a Population

Before proceeding to the explanation of the different genetic processes, we must first explain the coding of individuals. Each individual of the population is coded by a chromosome or a genotype [16]. The efficiency of the genetic algorithm will depend on the choice of the coding of a chromosome. In our case a chromosome is represented as chains of graphemes code obtained from the neural networks classification step.

3.4.2 Initial Population

The initial population is divided into two equal parts: It is produced by a random generation of n/2 chromosomes. In order to ensure the convergence of the GA, we insert the n/2 other chromosomes representing the graphemes having the best output recognition rate obtained by the neural networks as:

$$[\mathrm{Code}_{G1}(\mathrm{Rank}_1), \mathrm{Code}_{G2}(\mathrm{Rank}_2), \ldots \mathrm{Code}_{Gi}(\mathrm{Rank}_i), \ldots, \mathrm{Code}_{Gm}(\mathrm{Rank}_m)]$$

Where m is the number of graphemes composing the word, and $Rank_i$ is the rank of the chosen graphemes class in respect to the fuzzy probability of affectation issued by the neural network which is varied randomly from 1st to the 3rd rank.

3.4.3 Evaluation Function (Fitness)

The fitness function is the outcome of the comparison between two handwritten Arabic words (tested word and either a randomly selected word from initial population). GA tries to find a word (Character chains) which minimizes the evaluation function value (Fitness). The fitness value is calculated as follow:

1. Calculate the cost of grapheme instability = $1 - \mathrm{Max}\ (C_{GS}, C_{GNS})$
2. Calculate the cost of recognition uncertainty = $1 - C_{GA}$
3. Calculate the cost of grapheme shape variation = C_{GD}

The simultaneous inspection at equal level of importance of the criteria of the graphical stability of arranged forms, the recognition certainty of the modified visual form and its resemblance with the substituent visual form, is expressed by the calculation of the total fitness coefficient as the sum of the three precedent cost components.

The Tables 2, 3, 4 and 5 show the different values of the C_{GD} fitness component for each grapheme class: the value 0 means that the compared grapheme shapes are identical, while the value 1 is used when the graphemes shapes are completely different, value 0.25 means high similarity between the compared graphemes shapes and 0.75 is attributed if there is a little similarity of between them.

Table 2. Fitness function of beginning class

Graphemes code	Beginning nabra (ـبـ)	Ain (عـ)	Mim (مـ)	Kef (كـ)	Sad (صـ)	Haa (حـ)	Ha (هـ)	Qaf (قـ)	Lam (لـ)
Beginning nabra (ـبـ)	0	0.75	0.25	1	1	0.25	0.75	0.25	1
Ain (عـ)	0.75	0	0.75	0.25	0.75	0.75	0.75	0.25	1
Mim (مـ)	0.25	0.75	0	0.75	0.25	0.75	0.25	0.75	1
Kef (كـ)	1	0.25	0.75	0	1	0.25	1	1	1
Sad (صـ)	1	0.75	0.25	1	0	0.25	0.25	0.75	1
Haa (حـ)	0.25	0.75	0.75	0.25	0.25	0	0.75	0.75	1
Ha (هـ)	0.75	0.75	0.25	1	0.25	0.75	0	0.75	1
Qaf (قـ)	0.25	0.25	0.75	1	0.75	0.75	0.75	0	1
Lam (لـ)	1	1	1	1	1	1	1	1	0

Table 3. Fitness function of end class

Graphemes code	End nabra (ا)	Ain (ع)	Mim (م)	Kef (ك)	Sad (ص)	Haa (ح)	Ha (ه)	Qaf (ق)	Lam (ل)	Waw (و)	Raa (ر)	Yaa (ي)	Noun (ن)
End nabra (ا)	0	1	1	1	1	1	0.75	1	0.75	1	1	1	1
Ain (ع)	1	0	0.75	1	1	0.25	1	1	1	1	1	1	1
Mim (م)	1	0.75	0	1	1	0.75	1	1	1	1	1	1	1
Kef (ك)	1	1	1	0	1	1	1	1	1	1	1	1	1
Sad (ص)	1	1	1	1	0	1	1	0.75	1	1	1	0.75	0.25
Haa (ح)	1	0.25	0.75	1	1	0	1	1	1	1	1	1	1
Ha (ه)	0.75	1	1	1	1	1	0	1	1	1	1	1	1
Qaf (ق)	1	1	1	1	0.75	1	1	0	0.75	0.25	0.25	1	0.25
Lam (ل)	0.75	1	1	1	1	1	1	0.75	0	0.75	0.75	1	0.25
Waw (و)	1	1	1	1	1	1	1	0.25	0.75	0	0.25	1	0.25
Raa (ر)	1	1	1	1	1	1	1	0.25	0.75	0.25	0	1	0.25
Yaa (ي)	1	1	1	1	0.75	1	1	1	1	1	1	0	1
Noun (ن)	1	1	1	1	0.25	1	1	0.25	0.25	0.25	0.25	1	0

Table 4. Fitness function of isolated class

Graphemes code	Alif (ا)	Ain (ع)	Mim (م)	Kef (ك)	Sad (ص)	Haa (ح)	Ha (ه)	Qaf (ق)	Lam (ل)	Waw (و)	Raa (ر)	Yaa (ي)	Noun (ن)	Hamza (ء)	Daal (د)
Alif (ا)	0	1	1	1	1	1	1	1	0.75	1	1	1	1	1	1
Ain (ع)	1	0	1	1	1	0.25	1	1	1	1	1	1	1	1	1
Mim (م)	1	1	0	1	1	0.75	1	1	0.75	1	1	1	1	1	1
Kef (ك)	1	1	1	0	1	1	1	1	1	1	1	1	1	1	0.75
Sad (ص)	1	1	1	1	0	1	1	0.25	0.75	1	1	1	0.25	1	1
Haa (ح)	1	0.25	0.75	1	1	0	1	1	1	1	1	1	1	1	1
Ha (ه)	1	1	1	1	1	1	0	1	1	1	1	1	1	0.25	0.25
Qaf (ق)	1	1	1	1	0.25	1	1	0	0.75	0.75	0.75	1	0.25	1	1
Lam (ل)	0.75	1	0.75	1	0.75	1	1	0.75	0	0.75	0.75	1	0.75	1	1
Waw (و)	1	1	1	1	1	1	1	0.75	0.75	0	0.25	1	0.25	1	0.75
Raa (ر)	1	1	1	1	1	1	1	0.75	0.75	0.25	0	1	0.25	1	0.75
Yaa (ي)	1	1	1	1	1	1	1	1	1	1	1	0	0.75	1	1
Noun (ن)	1	1	1	1	0.25	1	1	0.25	0.75	0.25	0.25	0.75	0	1	1
Hamza (ء)	1	1	1	1	1	1	0.25	1	1	1	1	1	1	0	1
Daal (د)	1	1	1	0.75	1	1	0.25	1	1	0.75	0.75	1	1	1	0

Table 5. Fitness function of middle class

Graphemes code	Middle nabra (ـبـ)	Ain (ـعـ)	Mim (ـمـ)	Kef (ـكـ)	Sad (ـصـ)	Haa (ـحـ)	Ha (ـهـ)	Qaf (ـقـ)	Lam (ـلـ)
Middle nabra (ـبـ)	0	0.25	0.25	0.75	0.75	0.75	0.75	0.25	0.25
Ain (ـعـ)	0.25	0	0.25	0.75	0.25	0.75	0.25	0.25	1
Mim (ـمـ)	0.25	0.25	0	1	0.25	0.75	0.25	0.25	1
Kef (ـكـ)	0.75	0.75	1	0	1	0.75	1	1	0.75
Sad (ـصـ)	0.75	0.25	0.25	1	0	0.25	0.75	0.25	1
Haa (ـحـ)	0.75	0.75	0.75	0.75	0.25	0	0.75	0.75	1
Ha (ـهـ)	0.75	0.25	0.25	1	0.75	0.75	0	0.75	1
Qaf (ـقـ)	0.25	0.25	0.25	1	0.25	0.75	0.75	0	1
Lam (ـلـ)	0.25	1	1	0.75	1	1	1	1	0

3.4.4 Selection

To determine which individuals are more inclined to obtain the best results, a selection is operated. This operator selects among all individuals in a population who are most likely to reproduce. This selection is usually done over the fitness of individuals. The individual with the best coefficient of stability and the minimum cost of the fitness function will be selected.

3.4.5 Crossover

Process when new individuals are formed from parents. These new individuals are formed by making a cross between two parents with a higher probability for chromosomes possessing less fitness values. The main goal of the crossover stage is to

generate new solutions, which will contain the useful parts of both parent solutions and will have better fitness values.

3.4.6 Mutation

A random process when a gene changes value. This process plays a secondary role in the genetic algorithm, but it is still important. The rate of mutation in the respect to the total genetic operation is about 20%.

4 Experimental Results

To validate this method we have to test it on several words of the ADAB database (The Arabic handwriting Data Base). It contains more than 33,000 Arabic words, includes Tunisian town and village names, handwritten by 166 different writers [17, 18]. For graphemes classification, we have used a graphemes' database composed of more 10,000 samples. Table 6 below presents the results of graphemes recognition using MLP neural network. The average of the recognition accuracy was about 93.5% using the geometric parameters. The result was improved to 95.14% when we used the second vector of parameters which combines the geometric features with the Fourier descriptors modeling the handwritten trajectory shape. Test results show that the used feature vectors are very relevant for graphemes recognition.

Table 6. Graphemes classes' recognition rate

	Vector of geometric parameters	Vector combining geometric and Fourier descriptors parameters
Beginning grapheme	94.7%	95.3%
Middle grapheme	91.5%	92.7%
Isolated grapheme	97.4%	99%
End grapheme	94.9%	95.4%
Diacritic grapheme	89.3%	93.3%
Average	**93.5%**	**95.14%**

In order to calculate the recognition rates about the graphemes, characters and words; we have conducted a test of 300 words from ADAB database.

Table 7. Recognition error rate for different writing units

Used features vector	Graphemes error rate	Characters error rate	Word error rate
Vector of geometric parameters	6.5%	10%	30%
Combined geometric and Fourier descriptors vector	4.86%	8%	25%

The error rate obtained by our system for different writing units is illustrated by the Table 7 above.

Compared our work with the hybrid system proposed by [9] which is used the same ADAB database that also performed on online Arabic handwriting recognition. Although their system leads to an error rate about 2.5% which is lower than that obtained by our system, we note that their system works in a defined lexicon, while ours is designed for handwriting recognition application with free-lexicon.

5 Conclusion

We presented in this paper a new unlimited lexicon method for online Arabic handwriting recognition based on hybrid of neural networks and genetic algorithm. The given word is segmented into graphemes based on baseline extraction and Beta-Elliptic endpoints detection and then classified into five classes in respect to its position in the word trajectory. Each grapheme is then presented by a set of features vector combining geometric parameters and Fourier descriptors. These characteristics are then introduced in the input of an MLP neural network used for the recognition of the grapheme code. Finally, the GA is used to optimize the obtained chain of grapheme code in term of stability and certainty in a way to deduce the correspondent chain of character code. The results reveal that the hybrid features extracted by the Beta-elliptic model and Fourier descriptor of our method are useful in graphemes classification as well as the hybrid architecture MLP network and GA are efficient in characters code recognition. This proves that our method is promising. Its applicability to other scripts such as Latin and Persian would be a very interesting topic for future research.

References

1. Eraqi, H., Azeem, S.A.: An on-line Arabic handwriting recognition system: based on a new on-line graphemes segmentation technique. In: 2011 International Conference on Document Analysis and Recognition (ICDAR), pp. 409–413. IEEE (2011)
2. Ahmed, H., Azeem, S.A.: On-line Arabic handwriting recognition system based on hmm. In: 2011 International Conference on Document Analysis and Recognition (ICDAR), pp. 1324–1328. IEEE (2011)
3. Chaabouni, A., Boubaker, H., Kherallah, M., Alimi, A.M., El Abed, H.: Multi-fractal modeling for on-line text-independent writer identification. In: International Conference on Pattern Recognition, pp. 623–627 (2011)
4. Boubaker, H., Kherallah, M., Alimi, A.: New strategy for the on-line handwriting modelling. In: Proceedings of the 9th International Conference on Document Analysis and Recognition, Curitiba, Brazil, pp. 1233–1247 (2007)
5. El-Wakil, M.S., Shoukry, A.A.: On-line recognition of handwritten isolated Arabic characters. Pattern Recogn. **22**, 97–105 (1989)
6. Mezghani, N., Cheriet, M., Mitiche, A.: Combination of pruned Kohonen maps for on-line Arabic characters recognition. In: Proceedings of the Seventh International Conference on Document Analysis and Recognition (ICDAR 2003), pp. 900–904 (2003)

7. Alimi, A.M.: An evolutionary neuro-fuzzy approach to recognize online Arabic handwriting. In: Proceedings of the Fourth International Conference on Document Analysis and Recognition, pp. 382–386. IEEE (1997)
8. Boubaker, H., Tagougui, N., ElAbed, H., Kherallah, M., Alimi, A.M.: Graphemes segmentation for Arabic on-line handwriting modeling. J. Inf. Process. Syst. (JIPS) **10**(4), 503–522 (2014)
9. Tagougui, N., Boubaker, H., Kherallah, M., Alimi, A.M.: A hybrid NN/HMM modeling technique for online Arabic handwriting recognition. Int. J. Comput. Linguistics Res. **4**(3), 107–118 (2013)
10. Rokbani, N., Kherallah, M., Alimi, M.: On-line handwriting recognition system based on graph matching and genetic algorithm. In: International Conference on Machine Intelligence, Tozeur, Tunisia, pp. 959–963 (2005)
11. Elleuch, M., Tagougui, N., Kherallah, M.: Towards unsupervised learning for Arabic handwritten recognition using deep architectures. In: Arik, S., Huang, T., Lai, W.K., Liu, Q. (eds.) ICONIP 2015. LNCS, vol. 9489, pp. 363–372. Springer, Heidelberg (2015). doi:10.1007/978-3-319-26532-2_40
12. Tappert, C.C., Suen, C.Y., Wakahara, T.: The state of the art in on-line handwriting recognition. IEEE Trans. Pattern Anal. Mach. Intell. **12**(8), 787–808 (1990)
13. Bezine, H., Alimi, A.M., Derbel, N.: Handwriting trajectory movements controlled by a bêta-elliptic model. In: Proceedings of 7th International Conference on Document Analysis and Recognition, ICDAR 2003, January 2003, Edinburgh, United Kingdom, 3–6 August 2003, vol. 2, article no. 1227853, pp 1228–1232 (2003)
14. Dhieb, T., Ouarda, W., Boubaker, H., Ben Halima, M., Alimi, A.M.: Online Arabic writer identification based on beta-elliptic model. In: Proceeding of the 15th IEEE International Conference on Intelligent Systems Design and Applications (ISDA), Marrakech, Morocco, December 2015, pp. 75–79 (2015)
15. Singer, D.A., Kouda, R.: Application of a feedforward neural network in the search for Kuruko deposits in the Hokuroku district, Japan. Math. Geol. **28**, 1017–1023 (1996)
16. Holland, J.H.: Adaptation in Natural and Artificial Systems, 2nd edn. MIT Press, Cambridge (1992)
17. Boubaker, H., Chaabouni, A., Tagougui, N., Kherallah, M., Elabed, H., Alimi, A.M.: Off-line features integration for on-line handwriting graphemes modeling improvement. In: The 13th International Conference on Frontiers of Handwriting Recognition ICFHR 2012, 18–21 September 2012, Bari, Italy, pp. 69–74 (2012)
18. Chaabouni, A., Boubaker, H., Kherallah, M., El-Abed, H., Alimi, A.M.: Static and dynamic features for writer identification based on multi-fractals. Int. Arabic J. Inf. Technol. (IAJIT) (2013)

Intelligent Hybrid Algorithm for Unsupervised Data Clustering Problem

Amira Hamdi[1,2(✉)], Nicolas Monmarché[2], Mohamed Slimane[2], and Adel M. Alimi[1]

[1] REGIM-Lab.: REsearch Groups in Intelligent Machines, University of Sfax, ENIS, BP 1173, 3038 Sfax, Tunisia
{amira.hamdi7,adel.alimi}@ieee.org
[2] Computer Laboratory (LI), Polytech Tours, University of Tours, Tours, France
{Nicolas.monmarche,Mohamed.slimane}@univ-tours.fr

Abstract. Ant based algorithms have proved to be very efficient for solving real problems. These algorithms emphasize flexibility, robustness and decentralized control. Thus more and more researches are interested in this new way of designing intelligent systems in which centralization, control and preprogramming are replaced with self-organization, emergence and autonomy. In this context, many ant based algorithms have been proposed for data clustering problem. The purpose of this paper is to present a new intelligent approach for data clustering problem based on social insect metaphor and FCM algorithm.

Keywords: Data clustering problem · Swarm intelligence · Intelligent systems · Artificial ants

1 Introduction

In nature, a colony can coordinate its behavior and create complex patterns tasks without any direction and coordination between individuals. In this context, individuals in colony perform a work locally for global goal with sufficient flexibility as they are not controlled centrally. As examples of collective intelligent behaviors we can see ants that optimize food search, flock of birds that fly in a v-shaped formation, social spiders that build communal nest, termites that build collectively their sophisticated nest structure, honey bee swarms that cooperatively select their new nesting site, fireflies that flash its light in a wonderful pattern. These models have attached the attention of researches to propose several intelligent models to solve a wide range of complex problems. In this context, swarm intelligence is defined as a branch of artificial intelligence that is based on self-organization, emergence and stigmergy.

Recently, methods of swarm intelligence are used in data clustering problem. Clustering data is the process of classifying objects into groups of items that are similar between themselves and dissimilar to objects of other groups. Each object corresponds to a vector of M numerical values which correspond to the M numerical attributes. The relationships between objects are generated into a dissimilarity matrix in which rows and columns correspond to objects. As objects can be represented by points in numerical

A. Abraham et al. (eds.), *Proceedings of the 16th International Conference on Hybrid Intelligent Systems (HIS 2016)*, Advances in Intelligent Systems and Computing 552,
DOI 10.1007/978-3-319-52941-7_44

space, the dissimilarity between two objects can be defined as distance between the two corresponding point. Any distance can be used as a dissimilarity measures.

Many swarm-intelligence tools have been developed to solve clustering problems: Particle Swarm Optimization [10], Artificial Bee Colony [11], Firefly algorithm [12], Fish swarm algorithm [13] and Ant Colony Algorithm. A comprehensive review of the state of the art ant based clustering methods can be found in [9].

The rest of this paper is organized as follows. In Sect. 2, we present two swarm based clustering algorithms. A general description of our proposed clustering algorithm is given in Sect. 3. Experimental results and comparison studies are reported in Sect. 4. Finally concluding remarks are mentioned in Sect. 5.

2 Ant Clustering Algorithm

2.1 The Standard Ant Clustering Algorithm (LF)

One of the most attracting features of ant colony's behavior is its capability of organizing diverse components of the brood (eggs, larvae and nymphs) and arranging it in a concentric way in a same place (Fig. 1).

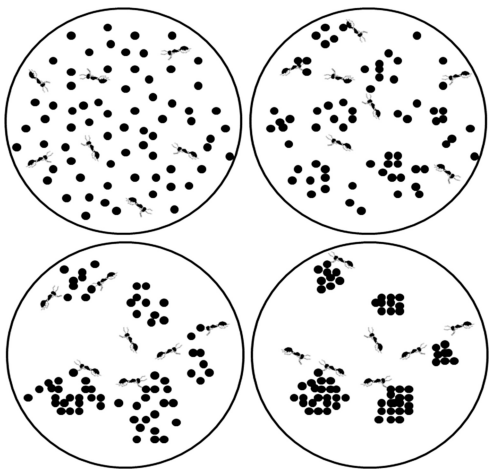


Fig. 1. Brood sorting by real ants.

In [1] *Lumer* and *Faieta* propose a distributed clustering method for data sets inspired by the collective sorting behavior of ants as initially proposed by *Deneubourg* [2]. In this model, objects have been laid out at random on *2-d* grid on which ants perform a random walk. The modeling of the rules used is relatively simple:

- A cell on the grid G is occupied at most one object o_i.
- A cell on the grid is occupied at most one ant.
- The number of objects exceeds the number of ants.
- The dimension of the grid is such that its number of cells exceeds the number of objects.

- An unloaded ant can pick an element on its current cell.
- A loaded ant c an drop a carried object according to the density of similar objects in its surrounding area.

– Rule1: when an ant encounters an element of the brood, the probability of capturing it is all the higher than if this element is more isolated.

$$P_{pick}(o_i) = \left(\frac{k_p}{k_p + f(o_i)}\right)^2 \tag{1}$$

Where k_p is a positive constant and f is the proportion of elements present in the neighborhood of the ant (f is somehow a measure of the local density).

When there are few objects in the neighborhood of the object coveted by the ant, $f << k_p$, which means that P_{pick} is close to *1* and the object is very likely to be picked up. Conversely, when the neighborhood is dense in elements, $f >> k_p$, and then P_{pick} is close to *0*;

– Rule2: when an ant carries an element of the brood, the probability of depositing it is all the higher than if the density of elements of the same type in the neighborhood is larger.

$$P_{drop}(o_i) = \begin{cases} 2f(o_i) & if \quad f(o_i) < k_{drop} \\ 1 & if \quad otherwise \end{cases} \tag{2}$$

Where k_d is a positive constant.

As can be seen, the local density function depends on the object o_i considered and on its position $r(o_i)$ on the grid. It is calculated as follows:

$$f(o_i) = \left\{ \frac{1}{s^2} \sum_{o_j \in R_S(r(o_i))} 1 - \frac{d(o_i,o_j)}{\theta} \quad ; \quad 0 \right\} \tag{3}$$

$f(o_i)$ is a measure of the average similarity of the object o_i with the objects o_j present in its neighborhood. θ is a scaling factor determining the extent to which the dissimilarity between two objects is taken into account. The normalizing term S^2 equals the total number of sites of in the local area of object o_i and introduces a density dependency in $f(o_i)$. Thereafter, the maximum value of f ($f = 1$) is reached if and only if all the sites in the neighborhood are occupied by identical elements. Figure 2 gives a grid example of *ACA* algorithm.

Where $d(o_i,o_j)$ measures the dissimilarity between the pair of objects (o_i,o_j) and it is given by:

$$d(o_i, o_j) = \left(\sum_{k=1}^{M} \left| x_{ik}, x_{jk} \right|^2 \right)^{\frac{1}{2}} \tag{4}$$

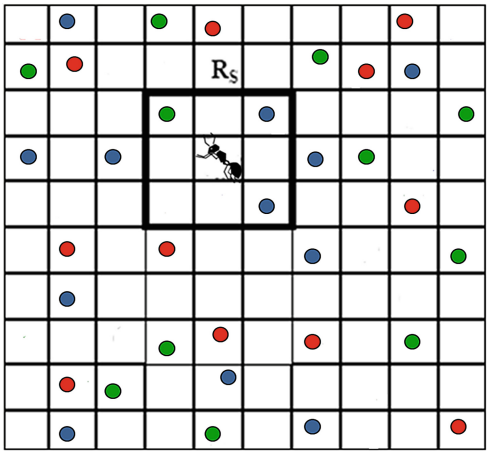


Fig. 2. Grid used in [1].

2.2 FlyAntClass Algorithm

In *LF* [1] and *AntClass* [3] algorithms, ants evolve arbitrarily on the grid. The proposed algorithm, that we have called *FlyAntClass* in [4], is motivated by our approach proposed in [6]. From Algorithm 2, we can see that we have proposed two kinds of moves for the ants: the random move and the intelligent move. In the first case, the ant selects a random direction among the neighbored possible ones. In the second case, described hereafter, the ant's move follows a local behavioral rule in a way that its displacement becomes intelligent.

In *FlyAntClass*, each object of the database is characterized by its features in the grid and by its neighborhood R_{os} of $s \times s$ cells which corresponds to all the objects around its current position that are given by the parameter V_{oi}.

In Algorithm 1, each ant a_k is characterized by its coordinates in the grid and by its velocity V_k which will be randomly generated at the beginning, then modified according to the move of ants by following a local rule, such that, in every iteration, ants carrying similar data move in the same direction.

For an ant a_k, characterized by a neighborhood R_{fs} of $s \times s$ cells around its current position, it is necessary to determine the parameter V_{fk} which corresponds to the set of ants situated in the nearby cells of a_k.

Two cases can occur here:

- If there is no ant in the neighborhood of a_k, then a_k is going to keep the same direction than previously.
- If V_{fk} not empty, then it is necessary to eliminate ants which are not carrying objects and calculate the sum of the influences of ants loaded with objects. This sum will be its new direction.

The basic idea is that the distance between two objects o_i *and* o_j on the grid has to converge towards an ideal distance which takes into account the similarity between the data. So, if the similarity between two objects carried by ants a_k and a_r is important,

and if these ants are too distant according to the ideal distance, it is thus necessary to move them closer. Conversely, if this similarity is weak and if ants are too close from each other, it is necessary to take them away. The parameters of our proposed algorithm are

Algorithm 1. General principle of *FlyAntClass* algorithm.

```
Initialize randomly the N objects {o_1, ..., o_N} on the grid G
For T = 1 to T_max do
  For all ant a_k do
    If a_k does not carry any object then
      If r(o_i) = r(a_k) then
        Compute f(o_i) and P_p(o_i)
        If ant a_k decides to pick up o_i according to
          probability P_p(o_i) then
            IntelligentMove()
            if blocking case is met then
              RandomMove()
            end if
          else
            RandomMove()
          end if
        else
          RandomMove()
        end if
      end if
      If ant a_k is carrying an object o_i then
        If r(a_k) is empty then
          Compute f(o_i) and P_d(o_i)
          if ant a_k decides to drop the object
           o_i on the cell r(a_k) according to
           probability P_d(o_i) then
            RandomMove()
          else
            IntelligentMove()
            if blocking case is met then
              RandomMove()
            end if
          end if
        else
          IntelligentMove()
        end if
      end if
    end for
  end for
```

Algorithm 2. Move strategies of ant in *FlyAntClass* algorithm.

Dertermine the neighbor V_{fk} of ant a_k
If $V_{fk} \neq 0$
then
$v_k(t+1) \leftarrow v_k(t)$
else
For all ant $a_r \in V_{fk}$ **do**
Let be o_i and o_j the two objects respectively carried by a_k and a_r
If $d(i, j) > d^*(i, j)$
then
$\beta(k, r) = 1$
else if $d(i, j) = d^*(i, j)$
then
$\beta(k, r) = 0$
else if $d(i, j) < d^*(i, j)$
then
$\beta(k, r) = -1$
end if
The influence of a_r on a_k is $V_{resultant}(k, r) = v_r + v_{ij} \times \beta(k, r)$
end for
Compute the sum of the influences: $V_{resultant}(k, .) = P_r$
end if

Table 1. Parameters of *FlyAntClass* algorithm.

Parameters	Definitions
A	Number of ants
V_k	velocity of ant k
V_{kr}	vector pointing from the ant a_k to the ant a_r
R_{fs}	neighborhood of ant k ($s \times s$ cells around the current position of ant k)
V_{fk}	set of ants situated in the neighbor of ant a_k
Ros	neighborhood of object o_i ($s \times s$ cells around the current position of ant o_i)
Voi	set of objects situated in the neighbor of ant o_i
d (i, j)	distance between two objects i and j
d*(i, j)	ideal distance between two objects i and j
$Sim_{threshold}$	similarity threshold
Sim_{mean}	average similarity
Sim_{max}	Maximal similarity

3 Proposed Methodologies

Results presented in [4] prove that ant-based clustering algorithms (*LF* and *FlyAntClass*) are not to be satisfactory approach for clustering in term of number of cluster. Besides it is important to note that the application of the heuristic is particular promising in case where the number of clusters in the data is not known. Most of the popular methods require an input parameter constituting the number of cluster. In another hand, [4] show that ant based clustering algorithms provide partition of data in which there are still some objects which are not assigned to any cluster when these algorithms stop. These objects are called in [3] "free objects". This corresponds to objects that are still carried by ants or to objects that are alone on the grid.

Thus, we propose in this paper a hybridization of *FlyAntClass* with alternative clustering methods that might therefore be a more rewarding to obtain more than marginal performance gains.

The *F-FlyAntClass* procedure consists of simply starting with unknown number of groups. The initialization part is identical to *FlyAntClass* [4]. We are able to show in Fig. 3 that the output of first step consists of classified and unclassified objects. Classified samples are represented by different symbol (red, green and blue). Unclassified ones are represented by black point.

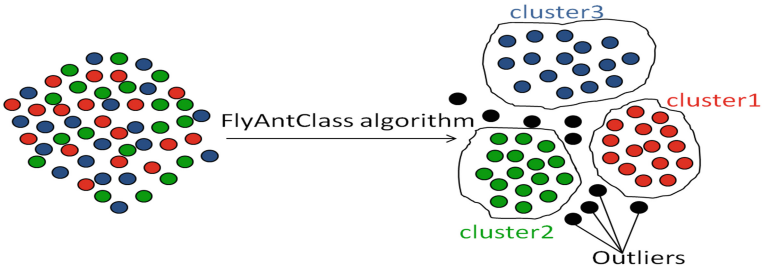


Fig. 3. Step 1 of *F-FlyAntClass* algorithm.

The result of the partition founded by *FlyAntClass* in the first step is given as input parameter to *FCM* algorithm in the second step. An element u_{ij} of partition matrix represents the grade of membership of object x_i in cluster c_j. Here, u_{ij} is a value that described the membership of an object i to class j.

We will initialize the partition matrix (membership matrix) given to *FCM* function by according *1* to classified object and *fz* to unclassified objects.

For the classified objects $u_{ij} = 1$ if object i in class j, otherwise $u_{ij} = 0$. For the unclassified objects, $u_{ij} = fz$ for all class j ($j = 1,2,3,\ldots,K$), described that, at the beginning an unclassified object i has the same membership to all class j. The sum of membership values across classes must equal one.

$$fz = \frac{1}{K} \tag{5}$$

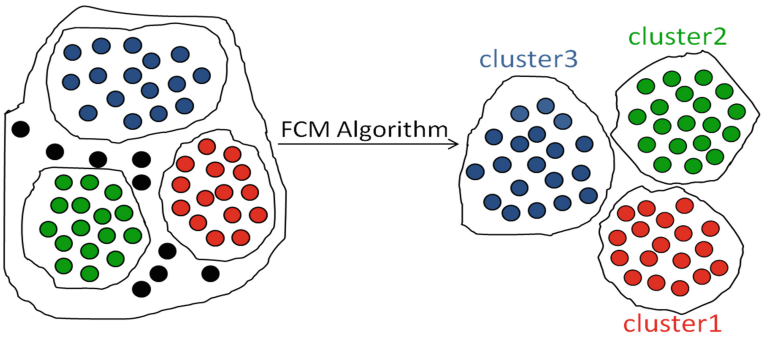


Fig. 4. Step 2 of *F-FlyAntClass* algorithm [14].

The output of the first step presented in Fig. 3 will is given as input parameter in the second step described in Fig. 4.

4 Experimental Results

4.1 Artificial and Real Data Sets

To evaluate the contribution of our method, we use several numerical datasets, including artificial and real databases from the *Machine Learning Repository* [5]. Concerning the artificial datasets, the database *Art1* and *Art3* are generated with *Gaussian* laws and with various difficulties. *Art*1 data set is the type of data most frequently used within previous work on ant based clustering algorithm [1, 3]. The general information about the databases is summarized in Table 2. For each data file, the following fields are given: the number of objects (*N*), the number of attributes (N_{Att}) and the number of clusters expected to be found in the datasets (*K*).

Table 2. Main characteristics of datasets used in our tests.

Datasets	N_{att}	N	K	Class distribution
Art1	2	400	4	(100, 100, 100, 100)
Art3	2	1100	4	(500, 50, 500, 50)
Thyroid	5	215	3	(150, 35, 30)
Pima	8	768	2	(500, 268)

The data are normalized in [0,1], the measure of similarity is based on *Euclidean* distance and the algorithms parameters used in our tests were always the same for all databases and all algorithms.

4.2 Evaluation Functions

The quality of the clustering results of the different algorithms on the test sets are compared using the following indices of cluster validity: *Number of clusters*, *silhouette* index, *Davies-Bouldin* index and *Calinski Harabasz* index.

The clustering *error Ec* measures the difference between the real partition and the obtained one in term of misclassification. It is defined as:

$$E_c = \frac{2}{N(N-1)} \times \sum_{(i,j)\in\{1,\ldots,N\}^2,\, i<j} e_{ij} \tag{6}$$

Where:

$$e_{ij} = \begin{cases} 0 \quad if \quad \big(C_k(o_i) = C_k(o_j) \ \wedge \ C_{k'}(o_i) = C_{k'}(o_j)\big) \vee \\ \qquad\qquad \big(C_k(o_i) \neq C_k(o_j) \ \wedge \ C_{k'}(o_i) \neq C_{k'}(o_j)\big) \\ 1 \quad else \end{cases} \tag{7}$$

The *silhouette* index mix ideas of both cohesion and separation. It determines which members do not fit very well. It is defined as:

$$s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}} \tag{8}$$

For each data o_i, let *a(i)* be the average dissimilarity of o_i with all other data within the same cluster and let *b(i)* be the lowest average dissimilarity of o_i to any other cluster, of which o_i is not a member.

$$a(i) = \frac{1}{n_k - 1} \sum_{\substack{o_{i'} \in C_k \\ i \neq i'}} d(o_i, o_{i'}) \tag{9}$$

$$b(i) = \min_{k' \neq k} \partial(o_i, C_{k'}) \tag{10}$$

$$\partial(o_i, C_{k'}) = \frac{1}{n_{k'}} \sum_{o_{i'} \in C_{k'}} d(o_i, o_{i'}) \tag{11}$$

A high *silhouette* value indicates that is well matched to its own cluster and poorly matched to neighboring cluster. It also can be applied at the level of dataset to determine which clusters are not very good. The maximal value indicates the best partition scheme of a dataset. The mean *silhouette* for a given cluster C_k is given by:

$$s_k = \frac{1}{n_k} \sum_{o_i \in C_k} s(i) \tag{12}$$

The mean of the mean *silhouettes* through all the clusters is called the *global silhouette* index and it is defined as:

$$Sil = \frac{1}{K'} \sum_{k=1}^{K} s_k \tag{13}$$

[7] defined the *Davies-Bouldin* criterion as:

$$DB = \frac{1}{K} \sum_{k=1}^{K} \max_{k \neq k'} \left\{ D_{k,k'} \right\} \tag{14}$$

The main idea is to compute for each cluster C_k the maximum value of the quantities $D_{kk'}$.

$D_{kk'}$ is the within to between cluster distance ratio for the *k-th* and the *k'-th*. It is defined as:

$$D_{kk'} = \left(\frac{\delta_k + \delta_{k'}}{\Delta_{kk'}} \right) \tag{15}$$

Where:

$$\partial_k = \frac{1}{n_k} \sum_{o_i \in C_k} d(o_i, C_k) \tag{16}$$

δ_k is the average distance between each point in the *k-th* cluster and the centroid of the *k-th* cluster. $\Delta_{kk'}$ is the *Euclidean* distance between the centroids of the *k-th* and clusters.

The smaller the *DB* value, the better the data partition.

Calinski-Harabasz index is defined in [8] as:

$$CH = \left(\frac{N - K}{K - 1} \right) \frac{SS_{BG}}{SS_{WG}} \tag{17}$$

The maximum value of *CH* index represents the best case with the optimal clustering partition. In fact, well defined clusters have a larger between cluster variance and a small within cluster variance.

SS_{BG} is the overall between cluster variance, $G\{k\}$ is the centroid of the cluster k, G is the overall mean of the sample data and SS_{WG} is the overall within cluster variance.

$$SS_{BG} = \sum_{k=1}^{K} n_k \left\| G^{\{k\}} - G \right\|^2 \tag{18}$$

$$SS_{WG} = \sum_{k=1}^{K} \sum_{o_i \in C_k} \left\| o_i^{\{k\}} - G^{\{k\}} \right\|^2 \tag{19}$$

4.3 Comparative Analysis

We denoted by δ_{Ec} the standard deviation of the clustering error. Bold face indicates the best result of algorithms. δ_{Sil}, δ_{DB} and δ_{CH} are respectively standard deviation of *Silhouette*, *Davies-Bouldin* and *Calinski-Harabasz* indices.

Tables 3, 4, 5 and 6 give results for *LF*, *FlyAntClass* and *F-FlyAntClass* algorithms. All Tables show mean, standard deviation, mode, worst and best value for 20 runs.

Table 3. Mean, deviation, mode, worst and best values of *error of classification* achieved by each algorithm (*LF*, *FlyAntClass* and *F-FlyAntClass*) over 20 trials and 1000 iterations on datasets presented in Table 2.

Datasets	Algorithm	Mean[δ_{Ec}]	Mode	Min	Max
Art1 [4,100]	LF	0.301 [0.012]	0.286	0.286	0.336
	FlyAntClass	0.294 [0.011]	0.278	0.278	0.314
	F-FlyAntClass	**0.223** [0.001]	0.225	0.218	0.225
Art3 [4,1100]	LF	0.421 [0.001]	0.419	0.419	0.424
	FlyAntClass	0.420 [6.483×10^{-04}]	0.419	0.419	0.421
	F-FlyAntClass	**0.405** [6.237×10^{-04}]	0.405	0.403	0.406
Thyroid [3,215]	LF	0.495 [0.012]	0.467	0.467	0.512
	FlyAntClass	0.514 [0.006]	0.516	0.505	0.534
	F-FlyAntClass	**0.470** [0.013]	0.462	0.449	0.496
Pima [2,768]	LF	0.541 [0.001]	0.539	0.539	0.543
	FlyAntClass	0.541 [0.001]	0.537	0.537	0.543
	F-FlyAntClass	**0.464** [0.004]	0.454	0.454	0.470

Table 4. Mean, deviation, mode, worst and best values of *silhouette* achieved by each algorithm (*LF*, *FlyAntClass* and *F-FlyAntClass*) over 20 trials and 1000 iterations on datasets presented in Table 2.

Datasets	Algorithm	Mean[δ_{Sil}]	Mode	Min	Max
Art1 [4,100]	LF	–0.813 [0.015]	–0.829	–0.764	–0.829
	FlyAntClass	–0.755 [0.043]	–0.810	–0.627	–0.810
	F-FlyAntClass	**0.427** [0.020]	0.388	0.468	0.388
Art3 [4,1100]	LF	–0.893 [0.006]	–0.906	–0.906	–0.878
	FlyAntClass	–0.873 [0.029]	–0.909	–0.909	–0.793
	F-FlyAntClass	**0.402** [0.016]	0.369	0.369	0.434
Thyroid [3,215]	LF	–0.689 [0.035]	–0.741	–0.741	–0.625
	FlyAntClass	–0.615 [0.082]	–0.706	–0.706	–0.418
	F-FlyAntClass	**0.201** [0.055]	0.266	0.110	0.312
Pima [2,768]	LF	–0.660 [0.012]	–0.682	–0.682	–0.639
	FlyAntClass	–0.616 [0.043]	–0.690	–0.690	–0.513
	F-FlyAntClass	**–0.246** [0.081]	–0.393	–0.393	–0.119

Table 5. Mean, deviation, mode, worst and best values of *Davies-Bouldin* achieved by each algorithm (*LF*, *FlyAntClass* and *F-FlyAntClass*) over 20 trials and 1000 iterations on datasets presented in Table 2.

Datasets	Algorithm	Mean[δ_{DB}]	Mode	Min	Max
Art1 [4,100]	LF	10.364 [2.809]	6.211	6.211	16.404
	FlyAntClass	20.585 [18.986]	8.528	8.528	97.558
	F-FlyAntClass	**0.888** [0.038]	0.825	0.825	0.961
Art3 [4,1100]	LF	33.082 [10.423]	21.855	21.855	57.807
	FlyAntClass	50.182 [50.625]	27.821	27.821	260.015
	F-FlyAntClass	**0.867** [0.020]	0.822	0.822	0.911
Thyroid [3,215]	LF	4.280 [0.663]	2.781	5.290	2.781
	FlyAntClass	6.360 [1.222]	4.326	4.326	8.629
	F-FlyAntClass	**1.215**[0.105]	1.100	1.054	1.385
Pima [2,768]	LF	4.434 [0.431]	3.763	5.159	*3.763*
	FlyAntClass	6.872 [0.411]	6.240	6.240	7.645
	F-FlyAntClass	**2.345** [0.299]	1.825	1.825	2.858

Table 6. Mean, deviation, mode, worst and best values of *Calinski-Harabaz* achieved by each algorithm (*LF*, *FlyAntClass* and *F-FlyAntClass*) over 20 trials and 1000 iterations on datasets presented in Table 2.

Datasets	Algorithm	Mean[δ_{CH}]	Mode	Min	Max
Art1 [4,100]	LF	1.568 [0.181]	1.234	2.042	1.234
	FlyAntClass	1.860 [0.354]	1.303	2.475	1.303
	F-FlyAntClass	**302.667** [9.420]	285.398	328.195	285.398
Art3 [4,1100]	LF	1.518 [0.129]	1.272	1.272	1.723
	FlyAntClass	1.574 [0.170]	1.176	1.176	1.869
	F-FlyAntClass	**700.288** [48.208]	593.695	593.695	792.206
Thyroid [3,215]	LF	4.164 [0.958]	2.672	2.672	5.769
	FlyAntClass	3.450 [0.849]	2.371	2.371	5.521
	F-FlyAntClass	**80.243**[13.737]	97.463	60.036	101.875
Pima [2,768]	LF	1.657 [0.080]	1.468	1.468	1.822
	FlyAntClass	1.497 [0.113]	1.286	1.286	1.716
	F-FlyAntClass	**34.319**[10.202]	21.140	21.140	57.408

As can be clearly seen in Tables 3, 4, 5 and 6, *F-FlyAntClass* outperforms both *LF* and *FlyAntClass* algorithms in term of all indices presented in Sect. 4.2.

In literature *Ant Clustering Algorithms* (*ACA*) were simulated during 10^6 iterations. Our proposed approaches (*FlyAntClass* and *F-FlyAntClass*) manage to have a better appreciation of the error of classification for only 10^3 iterations. Keeping in the mind that both *LF* and *FlyAntClass* algorithms provide partition of data in which there are still some objects which are not assigned to any cluster when these algorithm stop. These objects are called *Outliers* or so *free objects*. The advantage of *F-*

FlyAntClass is that all objects in dataset are assigned to their corresponding clusters. It successfully solves the problem of the existence of *Outliers*.

From Table 3, we can see that *F-FlyAntClass* perform better than *LF* and *FlyAntClass* for *Art3*, *Thyroid* and *Pima*. The average *error* classification is the best for *F-FlyAntClass* (*0.405* on *Art3*; *0.470* on *Thyroid*; *0.464* on *Pima*). The second best result is *0.420* on *Art3* for *FlyAntClass* and it is *0.495* on *Thyroid* for the *LF* algorithm. Whereas the worst result is *0.421* on *Art3* for *LF* algorithm and it is *0.514* on *Thyroid* for *FlyAntClass*.

The value of silhouette can vary between *–1* and *1*. Table 4 shows negative value of silhouette on all datasets for *LF* and *FlyAntClass* algorithms. Negative values are undesirable because these correspond to a case in which the average dissimilarity of oi with all other data within the same cluster is greater than the lowest average dissimilarity of oi to any other cluster, of which oi is not a member *(a(i) > b(i))*. For *F-FlyAntClass* the silhouette index has positive values on twice *Art3* and *Thyroid* datasets that indicate *a(i) < b(i)* and so an object oi has been assigned to an appropriate cluster.

Table 5 shows that *F-FlyAntClass* produces a partition of smallest *Davies-Bouldin* values on *Art3* (0.867), *Thyroid* (1.215) and *Pima* (2.345) datasets. The second smallest values are given by *LF* on *Art3* (*33.082*), *Thyroid* (*4.280*) and *Pima* (*4.434*). Worst values are given by *FlyAntClass* on all datasets. These values are respectively *50.182, 6.360* and *6.872*.

Results with *Calinski-Harabaz* index are presented in Table 6. The higher the value of *Calinski-Harabaz the* better is the solution. Best results are created with *F-FlyAntClass* on *Art3*, *Thyroid* and *Pima*. These larger values, which are respectively *700.288*, *80.243* and *34.319*, indicate more distinct clustering compared to both *LF* (*1.518*, *4.164* and *1.657*) and *FlyAntClass* (*1.574*, *3.450* and *1.497*).

5 Conclusion

Motivated by the problem of removing *Outliers*, this paper presented an improved *FlyAntClass* for data clustering problem. Our method, named *F-FlyAntClass*, is defined as an hybridization of *FlyAntClass*, with *FCM* algorithm. *F-FlyAntClass* has been tested with success on both artificial and real datasets and compared to those of the ant based clustering methods (*LF* and *FlyAntClass*). Those results are extremely encouraging in terms of the *error* of classification, *Silhouette*, *Davies-Bouldin* and *Calinski-Harabasz*. From now on, it is further suggested that a comparative study of ant based on clustering algorithms may be improved with different distance metrics.

Acknowledgment. The authors would like to acknowledge the financial support of this work by grants from General Direction of Scientific Research (DGRST), Tunisia, under the ARUB program.

References

1. Lumer, E., Faieta, B.: Diversity and adaptation in populations of clustering ants. In: Cliff, D., Husbands, P., Meyer, J., Wilson, S.W. (eds.) Proceedings of the Third International Conference on Simulation of Adaptive Behavior, pp. 501–508. MIT Press (1994)
2. Deneubourg, J.L., Goss, S., Franks, N., Sendova-Franks, A., Detrain, C., Chretien, L.: The dynamics of collective sorting: robot-like ant and ant-like robot. In: Proceedings of First Conference on Simulation of Adaptative Behavior: From Animals to Animats, pp. 356–365 (1991)
3. Monmarché, N.: On data clustering with artficial ants. In: Freitas, A.A. (ed.) AAAI 1999 & GECCO 1999 Workshop on Data Mining with Evolutionary Algorithms: Research Directions, pp. 23–26 (1999)
4. Hamdi, A., Monmarché, N., Slimane, M., Alimi, A.M.: FlyAntClass: intelligent move for ants based clustering algorithm. In: The 13th ACS/IEEE International Conference on Computer Systems and Applications (AICCSA 2016), Agadir, Morocco (2016)
5. Lichman, M.: UCI Machine Learning Repository. University of California, School of Information and Computer Science, Irvine, CA (2013). http://archive.ics.uci.edu/ml
6. Hamdi, A., Monmarché, N., Alimi, M.A., Slimane, M.: SwarmClass: a novel data clustering approach by a hybridization of an ant colony with flying insects. In: Dorigo, M., Birattari, M., Blum, C., Clerc, M., Stützle, T., Winfield, A.F.T. (eds.) ANTS 2008. LNCS, vol. 5217, pp. 411–412. Springer, Heidelberg (2008). doi:10.1007/978-3-540-87527-7_50
7. Davies, D. L., Bouldin, D. W.: A cluster separation Measure. IEEE Trans. Pattern Anal. Mach. Intell. **PAMI 1**(2), 224–227 (1979)
8. Calinski, T., Harabasz, J.: A dendrite method for cluster analysis. Commun. Stat. **3**(1), 1–27 (1974)
9. Hamdi, A., Antoine, V., Monmarché, N., Slimane, M., Alimi, A.M.: Artificial ants for automatic classification. In: Monmarché, N., Guinand, F., Siarry, P. (eds.) Artificial Ants: From Collective Intelligence to Real-life Optimization and Beyond, pp. 265–290. ISTE-WILEY (2010)
10. Omran, M., Salman, A., Engelbrecht, A.: Image classification using particle swarm optimization. In: Conference on Simulated Evolution and Learning, Singapore, vol. 1, pp. 370–374 (2002)
11. Karaboga, D., Ozturk, C.: A novel clustering approach: artificial bee colony (ABC) algorithm. J. Appl. Soft Comput. J. **11**(1), 652–657 (2011)
12. Senthilnath, J., Omkar, S.N., Mani, V.: Clustering using firefly algorithm: performance study. J. Swarm Evol. Comput. **1**(3), 164–171 (2011)
13. Xiao, L.: A clustering algorithm based on artificial fish school. In: 2nd International Conference on Computer Engineering and Technology (ICCET), pp. 766–769 (2010)
14. Hamdi, A., Monmarché, N., Slimane, M., Alimi, A.M.: Fuzzy rules for ant based clustering algorithm. Int. J. Adv. Fuzzy Syst. (2016)

Performance Analysis and Security Based on Intrusion Detection and Prevention Systems in Cloud Data Centers

Iman El Mir[1(✉)], Abdelkrim Haqiq[1], and Dong Seong Kim[2]

[1] Computer, Networks, Mobility and Modeling Laboratory FST, Hassan 1st University, Settat, Morocco
iman.08.elmir@gmail.com, ahaqiq@gmail.com
[2] Department of Computer Science and Software Engineering, University of Canterbury, Christchurch, New Zealand
dongseong.kim@canterbury.ac.nz

Abstract. Intrusion Detection Systems allow detection of unwanted attempts accessing which can violate security policy, manipulation or disabling of computer systems through the Internet. IDS plays a vital role in securing cloud data centers. This paper introduces an analytical model based on embedded Markov chain to analyze the packet processing using Bro IDPS (Intrusion Detection Prevention System). In this paper, we combine the preventive and detective rules to find a trade-off between network performance and security. We focus on how to enhance the detection engine and to predict the attack signatures taking into account of average service time, packet loss and blocking probability.

Keywords: Cloud computing · Intrusion detection systems · Markov chains · Performance · Security

1 Introduction

Intrusion Detection System (IDS) can monitor the behavior of network nodes timely, and find the suspicious behavior of nodes. Cloud computing applications and services are usually charged and the revenue is based on data or services used. Hence, the confidentiality and integrity of the data and timely availability of services is very important. Researchers have already investigated a lot of works in this area. Even with security that enables encryption and authentication, networks are vulnerable to a number of attacks aimed to disrupt the network. Hence, an Intrusion Detection System might be a significant option here to protect Cloud computing resources from some types of attacks that may not be possible to be addressed. An IDS is one possible solution to address a wide range of security attacks in cloud data center. An IDS is also referred to as a second line of defense, which is used for intrusion detection only. Once the intrusion is detected, the IDS raises an alarm to inform the controller for

A. Abraham et al. (eds.), *Proceedings of the 16th International Conference on Hybrid Intelligent Systems (HIS 2016)*, Advances in Intelligent Systems and Computing 552,
DOI 10.1007/978-3-319-52941-7_45

further actions. The functionality of IDS depends on three main components: data collection, intrusion detection and response to the intrusion. The data collection component is in charge for collecting the data from various data sources such as system audit data and network traffic data. The detection component is responsible for analysis of the collected data to detect the intrusions in a network. Thus, if any suspicious activity were detected in the network then initiate the response by the response module. There are two important techniques of IDS. One is rule-based IDS and the other is anomaly-based IDS [1,2]. Rule-based IDS is also known as signature-based IDS which is used to detect intrusions with the help of built-in signatures. Rule-based IDS can detect well-known attacks with great accuracy, and low false positive rate, but it is unable to detect new attacks for which the signatures are not present in intrusion database. Anomaly- based IDSs detect intrusion by matching traffic patterns or resource utilizations [3,4], based on the bases of normal behavior of the system. This technique can detect new and unknown attacks but it is still challengeable task due to the behavior of system which can be changed time to time. However, the disadvantage of this technique is that the false negative and false positive rates are high. IDPS require several detection techniques in integrated or separated forms in order to perform detection and prevention activities. Network based (NIDPS) and host based (HIDPS) are two major types of IDPSs. The IDPS systems contain a multiple sensors so as to filter and collect the information and to analyze them. The NIDPS sensors are responsible on networking packets processing, service and connectivity. However, the HIDPS sensors controls the system logs of the hosts, the memory utilization and CPU. The IDPSs offer a good technique in the first time in the collection and the analysis of the incoming data from different sources. These data should be well secured and protected from any attack which can be occured. Consequently, it's necessary to define the countermeasures and to detect the malicious activities source and its types. The rest of the paper is organized as follows: In Sect. 2, we present a preliminary discussion about IDS systems, their types, functions and their benefits in terms security and protection from malicious attacks. Our proposed system architecture is introduced in Sect. 3. As a use case, we consider Bro IDPS to prevent our cloud Data Center. We talk about its components and its efficienty. In Sect. 4 model Bro as IDPS behavior based on analytical model using embedded markov Chain. The performance evaluation of the IDPS analysis that optimize the fitness of cloud data center environment and the numerical results have been described in Sect. 5. Finally, we conclude our paper in Sect. 6 and provide details of the extension to real network environment along with scope for future work.

2 Related Work

Intrusion Detection and Prevention Systems (IDPS), are network security which its aim is to monitor network or system activities against malicious threats. The main operations are to report, block or to stop an intrusion. In order to model and analyze the efficiency of an IDPS, queueing theory is an important

methodology. The queueing system is given by a set of servers and a buffer which contains the incoming requests such as requests clients, network packets and HTTP requests. In [5], the authors analyzed the impact of security enforcement levels on the performance for an enterprise information system. They have developed an analytical model to study the IDPS performance and rules checking process. However, they presented different challenges in IDPS performance. Their results confirm a trade-off between system security and network performance. In the work [6], the authors studied the IDS performances. They quantified the detection rules in terms of benefits and costs. They analyzed IDS performance objectives. They evaluated the performance adaptation in real-time IDS through several experiments and simulations. Andreas Hes et al. [7] seek to measure the impact of IPS operation on network performance. They proposed an analytical model to describe the impact of a single router in the network. However, they formulated an optimization models to improve service security and to determine the optimal strategy. In [8], the authors proposed an analytical model using embedded Markov chain. Their aim is to evaluate and study the performance of rule-based firewalls. In [9], a survey on different intrusion detection techniques in cloud is presented. They have listed the intrusions which can menace the availability, integrity and the confidentiality of cloud services and resources. Through multiple simulations, they investigated the IDS and IPS as more efficient than firewalls to achieve high level of security especially in the next generation networks. In our first work [10], we have presented an analytic model for an Intrusion Tolerant Cloud Data Center. In particular, we have proposed to adopt the SCIT approach for a single VM and modeled the lifecycle of the VM. The model was implemented in SHARPE and the numerical results on system availability metrics were analyzed. The acquired results demonstrated that decreasing the exposure window will improve the intrusion tolerance of a SCIT-based Cloud Data Center. In [11,12], we have presented the previous security architectures, also discussed its advantages and inconveniences. In particular, we are focused on SCIT as an ITS architecture. So using Semi Markov process we model the preventive maintenance process on top of the existing intrusion tolerance mechanism. Quantitatively we have analyzed the system security using the measures such as system Availability, Mean Time To Security Failure and cost. The acquired findings demonstrated the feasibility of the proposed approach for improvement in system availability and reduction in downtime cost.

3 Proposed System Architecture

The proposed system architecture shown in the Fig. 1 based on IDPS as a defense system against malicious activities. We use IDS on each virtual machine in the cloud so as to control, analyze all incoming packet and to detect the threatened ones; we propose to combine the detection with prevention techniques in order to maintain a high level of integrity, confidentiality and availability for cloud data centers.

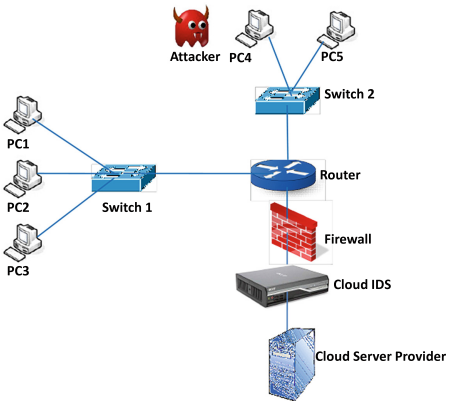


Fig. 1. Description of the system architecture.

3.1 Bro an Intrusion Detection System

Bro [13] is an open source network intrusion prevention and detection system (IDS/IPS) developed at LBNL and ACRI which provides a high performance intrusion detection system. Using Bro, we can optimize and analyze network traffic due to its benefits in terms high network level monitoring, low packet loss rate and definition of specific mechanism and policies. Bro is a useful technique which keeps all network connections and responds to network behavior patterns. It's defined as signature based systems. Its architecture is composed on three major components [14] as shown in Fig. 2.

- **Packet capture and filter:** built on Libpcap. It represents the kernel filters down high volume stream through packet capture library.
- **Event engine:** as protocol analysis which evaluates packets, maintains network connections states; If none already exists, it creates new state and generates events. So the events processing is according FIFO Queue.
- **Policy script interpreter:** executes scripts according to the policy language. All events should be transferred to the interpreter. In order to retrieve the compiled code for the corresponding handler and check the correspondence between the event values and handler arguments and finally interpret the obtained code.

Consequently, Bro IDS protects network and detects multiple-stage attacks. It works in two modes namely real-time processing and offline processing. In the first mode, it analyzes in real time the network traffic and filters the incoming traffic. Only the important ones are selected. It sends real time alerts based on policy. In the second mode, Bro is efficient to analyze the network traffic with more details but it doesn't filter all traffic, it considers only less traffic (Fig. 3).

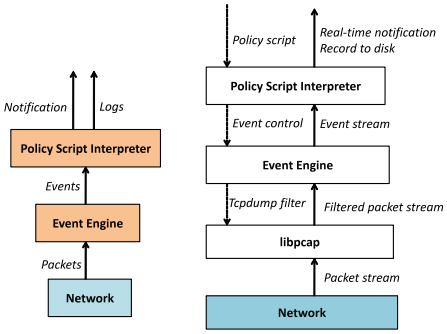


Fig. 2. Bro architecture.

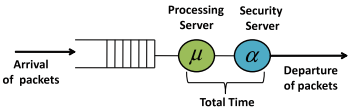


Fig. 3. Finite queueing model of IDPS.

4 Analytical Model for IDPS System

Bro as an efficient Intrusion detection prevention system can be modeled as two stage embedded Markov chain as depicted in Fig. 4. In this subsection, we dwell on the stochastic behavior of the proposed system. The aim of IDS system is to determine attack signatures, to enhance the event engine and to control the whole network. We consider Bro as IDPS system with two stages. The first stage is the processing server and the second stage corresponds to the security server. The finite queueing system was investigated as hypoexponential distribution with two stages. When a new request is coming, it will stay in the queue after it will be processed with mean service time μ. So in this first stage, the IDPS monitors the traffic processing knowing that only one packet should be processed in a time. Until it finishes, it can launch an other and so on. After, the IDPS detects the different traffic anomalies and triggers the second stage so as to treat the different signature of detected anomalies. However, the incoming packets can be adequately modeled as a Poisson process with rate λ. We have a finite queue. The buffer's size is $L-1$ ($L \neq 0$). The average service time in the first server is $1/\mu$ and the average service time in the second server is $1/\alpha$ (See Fig. 3). Service times for the two stages are exponentially distributed. To summarise our analytical model, we assume that the packet arrival is Poissonian, the services are exponential and packet's size is fixed. We evaluated the implementation of IDS in terms of throughput, blockage probability and packet loss. In this model, we consider the embedded Markov chain to represent the behavior of the two-stage service queueing system for Bro IDPS with a state space $S = \{(i,j), i \in [0,L], j \in \{0,1\}\}$ where i is the number of packets in the system and j represents

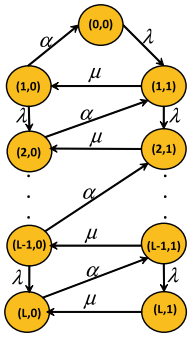


Fig. 4. Transition diagram model for Bro as IDPS

the stage the IDPS system is performing. The state $(0,0)$ denotes the special case where the system is idle. State transition diagram is depicted in Fig. 4. The steady state probability at state (i,j) is equal to $p_{i,j}$. The steady state equations are expressed as follow:

$$\lambda p_{0,0} = \alpha p_{1,0} \Longrightarrow p_{1,0} = \frac{\lambda}{\alpha} p_{0,0} \tag{1}$$

$$(\lambda + \alpha) p_{1,0} = \mu p_{1,1} \Longrightarrow p_{1,1} = (\frac{\lambda + \alpha}{\mu})(\frac{\lambda}{\alpha}) p_{0,0} \tag{2}$$

$$(\lambda + \mu) p_{1,1} = \lambda p_{0,0} + \alpha p_{2,0} \Longrightarrow p_{2,0} = (\frac{\lambda + \mu}{\alpha})(\frac{\lambda + \alpha}{\mu})(\frac{\lambda}{\alpha}) p_{0,0} - \frac{\lambda}{\alpha} p_{0,0} \tag{3}$$

$$(\lambda + \mu) p_{k,1} = \lambda p_{k-1,1} + \alpha p_{k+1,0}, \;\; k \geqslant 2 \tag{4}$$

$$(\lambda + \alpha) p_{k,0} = \mu p_{k,1} + \lambda p_{k-1,0}, \;\; k \geqslant 2 \tag{5}$$

$$p_{k,1} = (\frac{\lambda + \alpha}{\mu}) p_{k,0} - (\frac{\lambda}{\mu}) p_{k-1,0}, \;\; k \geqslant 2 \tag{6}$$

$$\mu p_{L,1} = \lambda p_{L-1,1} \tag{7}$$

$$\alpha p_{L,0} = \lambda p_{L-1,0} + \mu p_{L,1} \tag{8}$$

$$p_{L,1} = \begin{cases} \frac{\lambda}{\mu} p_{L-1,1} \text{ for } L > 1 \\ \frac{\lambda}{\mu} p_{0,0} \text{ for } L = 1 \end{cases} \tag{9}$$

$$p_{L,0} = \begin{cases} \frac{\lambda}{\alpha}(p_{L-1,0} + p_{L-1,1}) \text{ for } L > 1 \\ \frac{\lambda}{\alpha} p_{0,0} \text{ for } L = 1 \end{cases} \tag{10}$$

Furthermore, using the normalization condition, we determine $p_{0,0}$:

$$p_{0,0} + \sum_{k=1}^{L} (p_{k,0} + p_{k,1}) = 1 \tag{11}$$

5 Performance Parameters

5.1 Description of Performance Parameters

– **Outgoing Throughput:** The average number of packet per second in the system. It's defined as the rate at which packet finishes successfully being processed. Hence, we formulate the throughput β as follows:

$$\beta = \alpha \sum_{k=1}^{L} p_{k,0} \tag{12}$$

– **Probability of Blockage:** Is defined as the packet loss probability $P_{blockage}$. It is the probability of being in state (L,0), (L,1) where the new incoming packet will be refused. It is formulated as follows:

$$P_{blockage} = p_{L,0} + p_{L,1} \tag{13}$$

– **Average number of packets:** $\bar{N}$ in the system can be expressed as follows:

$$\bar{N} = \sum_{k=1}^{L} k \left[p_{k,0} + p_{k,1} \right] \tag{14}$$

Through, the Little formula, the average time spent in the system by the packet T_s is a function of $\bar{N}$ and β.

$$T_s = \frac{\bar{N}}{\beta} = \frac{1}{\alpha} \frac{\sum_{k=1}^{L} k(p_{k,0} + p_{k,1})}{\sum_{k=1}^{L} p_{k,0}} \tag{15}$$

– **The Average time spent in the queue by packet:** T_q is given by :

$$T_q = T_s - \bar{X} \tag{16}$$

where $\bar{X}$ denotes the mean service time. It's equal to the total service times of the two stages.

$$\bar{X} = \frac{1}{\mu} + \frac{1}{\alpha} \tag{17}$$

5.2 Numerical Results

The analytical model is evaluated numerically using Matlab tool. The numerical finding are shown in Figs. 5, 6 and 7. We have considered three different values for the mean service time of the two servers. We see that when the number of the rules increases ($\frac{1}{\mu} = 5\mu$s, $\frac{1}{\alpha} = 10\mu$s) in order to make our IDPS system more efficient with a high protection, the throughput becomes low. But in the last Figs. 5 and 6 we show that when the packet arrival rate increases and the mean service time in the two stages decreases, the average packet delay and the packet loss decreases. The reciprocal relationship between security and performance metrics. It's clear that when we involve more rule-checking using IDPS system we degrade the network performances, but we enhance the system security and detect and prevent the system from hostile attacks, which can compromise and produce fatal errors in the cloud data centers environment.

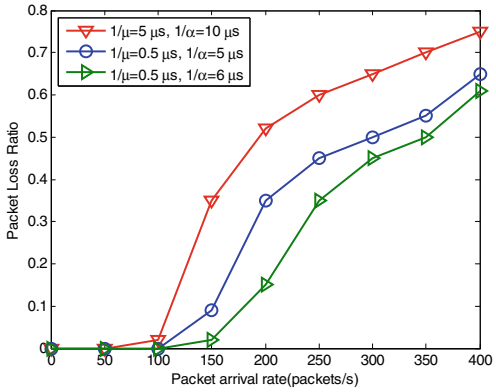


Fig. 5. Packet Loss Ratio versus packet arrival rate.

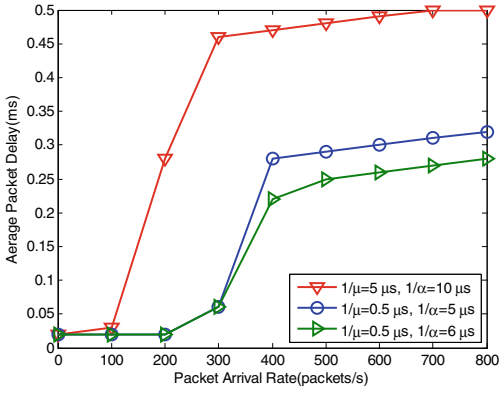


Fig. 6. Average time spent in the system per packet versus packet arrival rate

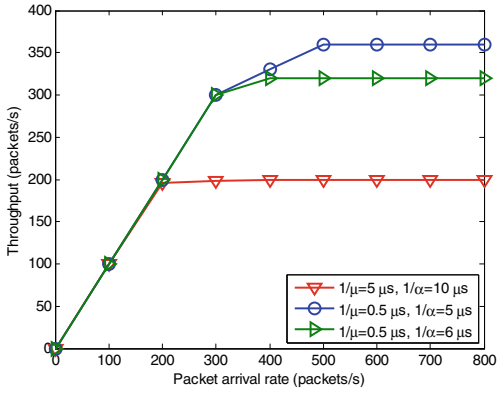


Fig. 7. Throughput versus packet arrival rate

6 Conclusion and Future Work

As cloud computing an emerging and sophisticated technology, many challenges were considered by IDPS systems in terms of attack signatures identification, detection engine improvement and system management. We have presented the IDPS system as mathematical model using embedded Markov chain in order to evaluate the impact of the IDPS configuration parameters on network performance. The key performance metrics evaluated in this paper are throughput, packet loss, blockage propability. However, we focus on finding the optimal strategy to tune IDPS parameters so as to control and manage its impact on network performance. On the other hand, In order to make this model more robust, we are going to study the hyperexponential distribution compared with the hypoexponential distribution of which the goal is to strike a balance between security assurance and network performance. Further, we seek to quantify and determine the vulnerability metrics and specify the transition points from one stage to another as well as to optimize system resources. Finally, in our future works we will conduct detection and prevention techniques based on our model with real data center scenarios and implement the proposed solution in OpenStack based cloud environment.

Acknowledgments. This research work is supported by the NATO Multi Year Project entitled Cyber Security Analysis and Assurance using Cloud-Based Security Measurement System with the code: SPS-984425.

References

1. Ambikavathi, C., Srivatsa, S.K.: Integrated intrusion detection approach for cloud computing. Indian J. Sci. Technol. **9**, 1–5 (2016)
2. Doelitzscher, F., Reich, C., Knahl, M., Passfall, A., Clarke, N.: An agent based business aware incident detection system for cloud environments. J. Cloud Comput. Adv. Syst. Appl. **1**, 1 (2012)
3. Langin, C., Rahimi, S.: Soft computing in intrusion detection: the state of the art. J. Ambient Intell. Humanized Comput. **1**(2), 133–145 (2010)
4. Patel, A., Taghavi, M., Bakhtiyari, K., Júnior, J.C.: An intrusion detection and prevention system in cloud computing: a systematic review. J. Netw. Comput. Appl. **36**(1), 25–41 (2013)
5. Alsubhi, K., Bouabdallah, N., Boutaba, R.: Performance analysis in intrusion detection and prevention systems. In: 12th IFIP/IEEE Integrated Network Management Symposium (IM), pp. 369–376. Trinity College Dublin, Ireland (2011)
6. Lee, W., Cabrera, J.B.D., Thomas, A., Balwalli, N., Saluja, S., Zhang, Y.: Performance adaptation in real-time intrusion detection systems. In: Wespi, A., Vigna, G., Deri, L. (eds.) RAID 2002. LNCS, vol. 2516, pp. 252–273. Springer, Heidelberg (2002). doi:10.1007/3-540-36084-0_14
7. Hess, A., Geerdes, H., Wessaly, R.: Intelligent distribution of intrusion prevention services on programmable routers. In: 25th IEEE INFOCOM, Barcelona, Spain. Citeseer (2006)
8. Salah, K., Elbadawi, K., Boutaba, R.: Performance modeling and analysis of network firewalls. IEEE Trans. Netw. Serv. Manage. **9**, 12–21 (2012)

9. Modi, C., Patel, D., Borisaniya, B., Patel, H., Patel, A., Rajarajan, M.: A survey of intrusion detection techniques in cloud. J. Netw. Comput. Appl. **36**(1), 42–57 (2013)
10. El Mir, I., Kim, D.S., Haqiq, A.: Security modeling and analysis of an intrusion tolerant cloud data center. In: Third World Conference on Complex Systems (WCCS), pp. 1–6. IEEE, Marrakech (2015)
11. El Mir, I., Kim, D.S., Haqiq, A.: Security modeling and analysis of a self-cleansing intrusion tolerance technique. In: 11th International Conference on Information Assurance and Security (IAS), pp. 110–116, IEEE, Marrakech (2015)
12. El Mir, I., Kim, D.S., Haqiq, A.: Cloud computing security modeling and analysis based on a self-cleansing intrusion tolerance technique. J. Inf. Assur. Secur. (JIAS) **11**, 273–282 (2016)
13. Paxson, V.: Bro: a system for detecting network intruders in real-time. Comput. Netw. **31**, 2435–2463 (1999)
14. Kacha, C.C., Shevade, K.K., Raghuwanshi, K.S.: Improved snort intrusion detection system using modified pattern matching technique. Int. J. Emerg. Technol. Adv. Eng. **3**(7), 81–88 (2013)

Multi-population Discrete Bat Algorithm with Crossover to Solve TSP

Wedad Al-Sorori(✉) and Abdulqader M. Mohsen(✉)

University of Science and Technology, Sana'a, Yemen
{w.alsrori,a.alabadi}@ust.edu
http://www.ust.edu

Abstract. Many meta-heuristic algorithms were proposed to solve several optimization problems. A new meta-heuristic bat algorithm (BA), inspired by the echolocation characteristics of micro-bats, has been extensively applied to solve continuous optimization problems. In addition, BA was also adapted to address combinatorial optimization problems. Unfortunately, like its basic version and other meta-heuristic algorithms, the adapted BA still suffers from some drawbacks such as slow speed convergence and easily trapping in local optima. We proposed a new variant of BA, called multi-population discrete bat algorithm (MPDBA), to solve traveling salesman problem (TSP). The validity of MPDBA was verified by comparative experiments using twenty TSP benchmark instances from TSBLIB. The experiments carried out show that MPDBA outperformed other state-of-art algorithms with respect to average and best solutions.

Keywords: Parallel hybrid bat algorithm · Crossover operator · Traveling salesman problem · Intensification and diversification · Discrete optimization problem

1 Introduction

Many meta-heuristic algorithms were inspired by natural phenomena such as individuals behaviors or biological evolutionary process and so on. These algorithms have attracted a great deal of attention due to their ability to find an optimal or a near to optimal solution for several optimization problems in a reasonable time [1].

TSP is one of the most famous discrete optimization problems that was proposed in the early 1930s by Karl Menger. Later, this problem was promoted by Hassler Whitney and Merrill Flood [2]. It was proved to be an NP-hard problem (NP stands for nondeterministic polynomial time). TSP popularity is due to being used as a benchmark for many optimization problems with the aim to find out the optimal solutions with minimum cost.

Bat algorithm is one of the recent meta-heuristic algorithms that has been successfully applied to solve several optimization problems including TSP [3]. To enhance the performance of basic BA, several hybrid variants were proposed to

A. Abraham et al. (eds.), *Proceedings of the 16th International Conference on Hybrid Intelligent Systems (HIS 2016)*, Advances in Intelligent Systems and Computing 552,
DOI 10.1007/978-3-319-52941-7_46

solve both continuous and combinatorial optimization problems. For example, to solve continuous numerical optimization problems, BA was hybridized other meta-heuristic algorithms such as [4–7] and methods such as [8–10]. Hybridization of BA with evolutionary operators such as mutation and differential evolution was also introduced [7,11,12] to solve UCAV path planning, image matching, numerical global optimization and constrained problems respectively.

In the same vein, basic BA was adapted to solve combinatorial problems such as hybrid flow shop scheduling problems, feature selection problem, scheduling workflow applications in cloud, multidimensional knapsack problem, permutation flow shop scheduling problem, community detection problem in the network, path relinking and capacitated vehicle routing problem in [13–19] respectively. To address TSP, Yassine Saji et al. proposed a discrete version of BA [20,21]. Then Jihen Amara et.al proposed another discrete BA variant [22]. Recently, Eneko Osaba et al. introduced an improved discrete bat algorithm for both symmetric and asymmetric TSP called IBA [23]. In addition to hybridization, parallelization was also introduced to enhance basic BA. BA was parallelized to solve numerical optimization problems [24]. Similarly, a communication strategy for a hybridized BA with particle swarm optimization [6] and BA with artificial bee colony [5] were also introduced. To the best of our knowledge, there were only two parallel variants of BA were proposed to solve combinatorial optimization problem. The first investigated a multi-population cooperative version of bat algorithm for association rule mining [25]. The second proposed a multi-population bat algorithm to solve classification problem of an artificial neural network (ANN) [26].

In this paper, we proposed a new multi-population variant of discrete BA to solve TSP with three main contributions. The first one is the embedding of local search techniques (2-opt and 3-opt) to speed up the convergence toward the optimal solution. The second is the introduction of the crossover operator to exploit the current search region efficiently. The third is the exploitation and utilization of the concepts of the parallel island model of genetic algorithms to maintain the diversity and avoid being trapped in local minima.

The rest of this paper is organized in five sections as follows. Section 2 briefly introduces the standard BA, IBA and crossover operation. Section 3 describes the proposed algorithm to solve TSP. Section 4 demonstrates the experimental results, statistical analysis and discussion. Section 5 presents the conclusion and future work.

2 Preliminaries

In the following subsections, we briefly review the bat algorithm, its IBA discrete variant and crossover operation.

2.1 Bat Algorithm

Bat algorithm was initiated by Yang 2010 [3] as a relatively new meta-heuristic to solve hard optimization problems. It has received a great deal of attention

regarding its potential as an optimization algorithm for solving a wide range of optimization problems. An overview of BA, its inspiration idea, its variants and application domains were shown in [27]. The main steps of BA are illustrated as follows:

Step1 (Initialization): An initial population $X = [x_1, x_2, \ldots, x_n]$ of n bats is generated randomly. Each bat i in the population represents a potential location (solution of the problem under consideration) with random rate of pulse r_i, random loudness A_i and initial frequency f_i; each of which is evaluated using fitness function. The solutions evolve through successive iterations in which bats move from their initial positions toward global best position(s).

Step2 (Generation of New Solutions): During each iteration, every bat i updates its frequency f_i, velocity v_i and position (solution) x_i at iteration t according to Eqs. 1, 2 and 3 respectively.

$$f_i = f_{min} + (f_{max} - f_{min})\beta \tag{1}$$

$$v_i^t = v_i^{t-1} + (x_i^{t-1} - x_*), \tag{2}$$

$$x_i^t = x_i^{t-1} + v_i^t, \tag{3}$$

where β is a random vector in the range of $[0, 1]$ drawn from a uniform distribution; and x^* is the current global best location (solution) among all solutions in the population.

Step3 (Local Search): After that, with some probability of pulse rate r_i, a solution is selected among the best solutions and random walk is applied to generate a local solution around the selected one according to Eq. 4:

$$x_{new} = x_{old} + \varepsilon A_i^t, \tag{4}$$

where the random number ε is a scale factor drawn from $[-1, 1]$ and A_i^t is the average loudness of all bats. Then, the new solutions will be accepted if they are improved or by flying randomly with some probability depending on A_i.

Step4 (Loudness and Pulse Emission): If the solution is accepted, the pulse emission rate r_i is increased and the loudness A_i is decreased like what happens when natural bat finds its prey. Mathematically, this is defined using Eqs. 5 and 6:

$$A_i^{t+1} = \alpha A_i^t, \tag{5}$$

$$r_i^{t+1} = r_i^0[1 - \exp(-\gamma t)], \tag{6}$$

where α and γ are constants. Parameter α plays a similar role as the cooling factor in the simulated annealing algorithm to control the convergence rate of BA. It is worthy to note that loudness A_i and pulse emission rate r_i are the most important parameters of BA in which pulse emission rate increases the population diversity since it causes update of position(s). Hence, more new positions can be explored and, in turn, r_i controls the improvement of the best solution(s). In a similar vein, loudness strengths the local search and directs bats to find better solutions. It also influences the acceptance of the best solution(s).

Step5 (Finding the Global Best Solution): Finally, rank the solutions and find the current global best solution(s).

Steps 2–5 continue until the termination condition is satisfied.

2.2 An Improved Bat Algorithm (IBA) for TSP

Eneko Osaba et al. presented an improved discrete version of bat algorithm called IBA [23]. It was used to solve TSP, but it failed to reach the optimal solution in the most of instances.

IBA used the same philosophy of the basic BA parameters r_i and A_i. However, velocity parameter v_i was calculated according to Eq. 7:

$$v_i^t = random[1, HammingDistance(x_i^t, x^*)], \tag{7}$$

where v_i of a bat i at iteration t is a random number between 1 and the difference between the current bat position x_i and the best bat position x^* in the population. The difference is calculated by the hamming distance which represents the number of non-corresponding elements in the solution. The modification in IBA was based on the movement behavior of bats in which all bats were given some kind of intelligence so that each bat moves differently depending on how far it is from the best bat of the population. In this way, when one bat moves, it will first examine its velocity v_i^t. If the velocity value is less than the half of the cities, the bat i performs short move using 2-opt local search; otherwise, it performs long move using 3-opt.

Notably, the IBA structure tends to intensification due to the use of local search techniques, such as 2 and 3 opt. Therefore, IBA still suffers from some drawbacks such as premature convergence and easy trapping in local optima.

2.3 Crossover Operation

Crossover operation is inspired by evolutionary algorithms (EA). It works with selecting two solutions as parents from the existing population and creates two new solutions which are more similar to their parents. Thus, crossover operation allows intense search in the neighborhood and, hence, exploits the solution space for local search efficiently. In the literature, crossover operator has different implementation such as single point crossover, two-point crossover, uniform crossover and partially matched or mapped crossover (PMX), etc. PMX crossover operator proposed by Goldberg and Lingle [28] is the most commonly used for TSP. In PMX, two solutions are selected and two crossover points are chosen randomly. The part of solutions between the two crossover points gives a matching selection which affects cross through position-by-position exchange operations.

3 The Proposed Algorithm

The multi-population discrete BA model (MPDBA) is a parallel BA variant inspired by the island model in GA which in turn was originally inspired by

nature. Two main advantages are gained from the parallel BA model. The first is to improve the solution of the problem and maintain the diversity by preventing premature convergence that faces basic BA. The second is to reduce the convergences time of BA processes by evaluating the bats in the multi populations concurrently. The cooperative model is a logical structure which can be implemented on many different architectures. However, the model in this paper consists of several populations which are distributed into the processors available in the multi-thread architecture. Each population searches for the best bats independently of the remaining populations and exchanges random bats with others from time to time.

The detailed explanation of the proposed MPDBA variant is shown in the following steps.

Step 1 (Parameters Initialization): Initialize the parameters of MPDBA that control the algorithm during the updating step. These parameters are defined as follows.

i **Basic BA Parameters**: There are four important parameters, the loudness, pulse emission rate, velocity parameter and population. While α and γ are constants.

ii **Number of Populations (NP)**: NP is the number of populations that work in the model. Generally, each population has the same number of bats and as such every population is assigned to a separate thread in a multi-threading system.

iii **Exchange Interval (EI)**: EI determines the number of iterations that should be taken before exchanging bats from one population to another. The intermediate length of EI is expected to provide sufficient balance between intensification and diversification.

iv **Exchange Rate (ER)**: ER represents the number of bats that should be transferred from one population to another. ER has a range of values between 0 and 1. The medium ER may guarantee the balance between the diversification and intensification.

v **Exchange Strategy (ES)**: ES determines the way of selecting the source and destination populations for exchanging.

vi **Exchange Policy (EP)**: EP determines how the bats for exchange are chosen from the source population, and how they are replaced in the destination population.

vii **Number of Iterations (NI)**: NI determines the number of iterations required to search the optimal solution. The algorithm will terminate when the optimal solution is found or the maximum number of iterations is reached.

Step 2 (Population Initialization): Initialize each population with random values for each bat from the possible range of values. This step ensures the variety of solutions. For TSP, each population is initialized as a matrix with (NxM), where N is the number of bats (tours) and M is the number of components (cities). Then evaluate the length of the routes completed by

each bat in the population according to Eq. 8.

$$Tour\_length = \sum_{i=1}^{n} d_{i\pi(i)}, \tag{8}$$

Where $d_{i\pi(i)}$ is the distance between city (i) and $\pi(i)$; the city on the tour follows city (i).

Step 3 (Evaluate Each Individual inside Population): Compute bats velocities and evaluate each bat in the population using the fitness function. The velocity v_i is computed according to Eq. 7.

Algorithm 1: Multi-Population Discrete Bat Algorithm (MPDBA)

```
1  begin
2      Input:objective function f(x);
3      Output: the best solution x*;
4      t=0;
5      for m = 1 to No_of_Populations do                    /* For each bat population in the model do */
6          Initialize the population p[m] = [x_1, x_2, ...., x_n] ;
7          for i = 1 to No_of_bats do                       /* For each bat in population[m]  */
8              Calculate the objective function f(x_i) for bat's position x_i;
9              Initialize the pulse rate r_i,velocity v_i and loudness A_i;
10         end
11         find the best bat's position x* in p[m];
12     end
13     while termination criterion not reached do
14         for m = 1 to No_of_Populations do                /* For each population  */
15             for each bat i in the population do
16                 Generate new solution;
17                 crossover(x_i,x*)
18                 if v_i^t < n/2 then
19                     x'_i = 2 - opt(x_i^{t-1}, v_i^t)
20                 else
21                     x'_i = 3 - opt(x_i^{t-1}, v_i^t)
22                 end
23                 if rand > r_i then
24                     Select one solution among the best ones;
25                     Generate a new bat position, (x''_i), selecting the best neighbor around the chosen
                         bat using the 2-opt or the 3-opt;
26                 end
27                 if f(x'') < f(x') then
28                     x_i=x''_i;
29                 else
30                     x_i= x'
31                 end
32                 if rand < A_i and f(x_i) < f(x_i - x*_n) then
33                     Accept the new solution;
34                     Increase r_i and reduce A_i;
35                 end
36             end
37             Evaluate each bat in p[m];
38             find the current best and current worst in p[m];
39         end
40         find the current best among all populations;
41         if t Mod exchange_interval==0 then
42             for m = 1 to No_of_Populations do
43                 Select fraction of bat from source p[m] equal to ER;
44                 Randomly, determine an index k of the destination population which has not been
                     selected before;
45                 Send selected bats to p[k] based on ET;
46                 Remove fraction of bats from p[k] equal to the incoming bats based on EP;
47             end
48         end
49         t=t+1 ;                                          /* t is the iteration number  */
50     end
51     output the best solution x* among all populations;
52 end
```

Step 4: (Population Updating): For each bat i in the model, generate a new better bat position (solution) by firstly performing the crossover operation between the current bat position x_i and the current global best bat position x^* in the same population. After that, the fitness of the current bat position is compared with those of the two offspring produced by crossing. The best position is chosen as the new position of current bat. By using the crossover operator, BA can make use of the others advantage to avoid being trapped into local optima. Secondly, each bat i moves according to its velocity and the distance between it and the best one in the population. This movement can be done by using either 2-opt or 3-opt local search procedures. Then the fitness function of the new bat is calculated accordingly.

Step 5 (Loudness and Pulse Emission): If the solution is accepted, r_i is increased and A_i is decreased using Eqs. 5 and 6.

Step 6 (Finding the Global Best Solution): Rank the solutions and find the current global best solution(s).

Step 7 (Bats Exchanging): If the EI is reached, determine the source and the destination populations according to ES to exchange number of bats equals to the ER between the two populations based on EP.

Step 8: (Termination Criterion Checking): The MPDBA algorithm is terminated when the optimal solution is found or the maximum number of iterations is reached. Algorithm 1 summarizes the structure of MPDBA.

4 Experimental Results

In this section, the performance of the proposed algorithm was evaluated in terms of the achieved computational results. Different experiments were conducted for the evaluation using Intel core-i5 machine. The evaluation was accomplished using different symmetric TSP standard instances, with different length obtained from TSPLIB (http://comopt.ifi.uni-heidelberg.de/software/TSPLIB95/). All instances included in TSPLIB have been already examined in the literature

Table 1. MPDBA parameter setting

Parameter	Tested values	Optimum value
Iterations number	100 500 1000	500
Number of populations (threads)	5 10 15	10
Exchange Interval (EI)	10 30 50	30
Exchange Rate (ER)	0.10 0.20 0.30	0.20
Exchange topology	Fully or partially connected graph	Fully connected graph
Exchange policy	Random or best selection	Random selection and substitution with the worst

Table 2. Computational results of BA-opt, BA-xover and MPDBA. All of the results were taken from 10 runs. The best results are given in bold.

Instances	Optimal	BA-opt		BA-xover		MPDBA	
		Avg.	*Best*	*Avg.*	*Best*	*Avg.*	*Best*
Oliver30	420	**420**	**420**	**420**	**420**	**420**	**420**
berlin52	7542	**7542**	**7542**	**7542**	**7542**	**7542**	**7542**
St70	675	676.8	**675**	678.3	**675**	**675**	**675**
Eil51	426	427.1	**426**	427.1	**426**	**426**	**426**
Eil76	538	544.6	**538**	542	**538**	**538**	**538**
Eil101	629	636.2	631	636.4	631	**629.2**	**629**
KroA100	21282	21335.3	**21282**	21311.2	**21282**	**21282**	**21282**
Krob100	22141	22245.4	**22141**	22200.2	22193	**22141**	**22141**
Kroc100	20749	20769.3	**20749**	20778	**20749**	**20749**	**20749**
Krod100	21294	21428.6	21309	21348.6	**21294**	**21295.5**	**21294**
Kroe100	22068	22143.4	**22068**	22121.7	**22068**	**22073.4**	**22068**
Pr107	44303	44432.4	**44303**	44401.6	**44303**	**44303**	**44303**
Pr124	59030	59065.7	**59030**	59139.1	**59030**	**59030**	**59030**
Pr136	96772	97654.8	97007	97603.3	96861	**96775.1**	**96772**
Pr144	58537	58676.4	**58537**	58631.5	**58537**	**58537**	**58537**
Pr152	73682	74354.2	73818	74151.8	73840	**73695.6**	**73682**
Pr264	49135	49410	**49135**	49401.1	**49235**	**49139.5**	**49135**
Pr299	48191	48903.5	48609	48687.6	48372	**48194.4**	**48191**
Pr439	107217	109232.5	108524	109058.6	108387	**107217.6**	**107217**
Pr1002	259047	267616.1	266302	266719.8	265459	**259495**	**259495**

and their optimality results were used to validate the efficiency of the proposed algorithms according to the best and average values. The results were collected after running the experiments 10 times for each instance. Each population in MPDBA was initialized with 50 bats each with initial random values in the range 0.7–0.1 and 0.0–0.4 for the loudness and pulse parameters respectively. Alpha and gamma parameters were set to 0.98. The maximum number of iterations was 500. These BA parameter values were mentioned in [23]. However, the additional MPDBA parameters were tuned and tested to get their optimal values as shown in Table 1.

Table 2 shows the improvement of the computational results using twenty TSP instances with respect to the best and average solution. As it can be seen from the tabulated values, BA-xover was able to obtain the optimal solution as BA-opt in 12 instances. BA-xover can also achieve the optimal solution in Krod100 where BA-opt failed to achieve. Regarding the average results obtained, both algorithms obtained the optimal solution in two small instances Oliver30 and berlin52. They were identical to obtain near to optimal solution in Eil51. BA-opt achieved slightly better results in only two instances Eil101 and Kroc100.

BA-xover is superior over BA-opt in the remaining 15 instances. This may be because adding the crossover operation can enhance the learning capability of the algorithm from the previously detected promising areas and exploit the current search area efficiently. In the same vein, MPDBA achieved as the optimal solution for all instances, with respect to the best solution. Furthermore, MPDBA outperformed BA-xover in almost 90% of all tested instances, with respect to average solution. This superiority of MPDBA is referred to the utilization of the parallelized multi-population model in the search process of the algorithm which fosters the diversity in case the solutions trapped into the local optima.

MPDBA was also compared with the recent discrete BA algorithms, IBA [23], and the evolutionary simulated annealing, ESA [29]. Table 3 shows the comparison results of MPDBA with IBA and ESA, with respect to the best solution, average solution, percentage deviations of the average solution PD_{avg} and percentage deviations of the best solution PD_{best}. PD_{avg} and PD_{best} were defined in Eqs. 9 and 10 respectively.

$$PD_{avg} = \frac{(average\_solution - best\_known\_solution)}{best\_known\_solution} \times 100 \tag{9}$$

$$PD_{best} = \frac{(best\_solution - best\_known\_solution)}{best\_known\_solution} \times 100 \tag{10}$$

As seen from Table 3, the obtained results of MPDBA were better than those of IBA and ESA. For example, the average results obtained for the Krob100 instance by IBA and ESA were 22506.4 and 22602.2 respectively. While MPDBA gained the optimal solution i.e. 22141 in all runs. Accordingly, the relative PD_{avg} was 1.650, 2.083 and 0.0000 for IBA, ESA and MPDBA respectively. Compared to the published results in the literature, it is clear that MPDBA has obtained closer results to optimum in most of the instances. In spite of the superiority of MPDBA over IBA and ESA in terms of finding the best solution, solving large-scale instances required more computing time. In addition, PD_{avg} of MPDBA for bigger instances was greater than that of small instances. This may be because of the fact that the complexity increases exponentially as the number of cities increases. One of the reasons that MPDBA does not obtain the optimal solutions in some of the large-scale instances may be because it gets stuck into the local minima, and thus, further improvements are required. In general, the average and best solutions obtained by MPDBA were satisfactory, and significantly better than those found by both IBA and ESA. The numerical results show that MPDBA was able to solve small and large size problems better than IBA and ESA. It is worthy to mention that incorporation of the local search techniques, crossover operator and parallel processing model to BA is effective, promising and improves the results significantly.

Statistical tests have been conducted using the obtained results for rigorous and fair conclusions. The tested variables, algorithms, distribution was not normally distributed, thus the nonparametric test should be used. Since there were 3 variables, i.e. MPDBA, IBA and ESA, a multiple comparison test should be used. Firstly, the Friedman Test tests the null hypothesis that there was no significant differences among all tested algorithms. In Table 4, the resulting

Table 3. Computational results of MPDBA in comparison with IBA and ESA. All of the results for IHDBA were taken from 10 runs. The best results are given in bold.

Instances	Optimal	IBA				ESA				MPDBA			
		Avg.	*Best*	*PD_avg*	*PD_best*	*Avg.*	*Best*	*PD_avg*	*PD_best*	*Avg.*	*Best*	*PD_avg*	*PD_best*
Oliver30	420	**420**	**420**	0	0	**420**	**420**	0	0	**420**	**420**	0	0
berlin52	7542	**7542**	**7542**	0	0	**7542**	**7542**	0	0	**7542**	**7542**	0	0
St70	675	679.1	**675**	0.607	0	682.1	**675**	1.052	0	**675**	**675**	0	0
Eil51	426	428.1	**426**	0.493	0	431.6	**426**	1.315	0	**426**	**426**	0	0
Eil76	538	548.1	539	1.877	0.186	553.7	546	2.918	1.487	**538**	**538**	0	0
Eil101	629	646.4	634	2.766	0.795	658.4	650	4.674	3.339	**629.2**	**629**	0.032	0
KroA100	21282	21445.3	**21282**	0.767	0.000	21481.7	21282	0.938	0.000	**21282**	**21282**	0.000	0
Krob100	22141	22506.4	22140	1.650	−0.005	22602.2	22202	2.083	0.276	**22141**	**22141**	0.000	0
Kroc100	20749	21050	**20749**	1.451	0.000	21170.4	**20749**	2.031	0.000	**20749**	**20749**	0.000	0
Krod100	21294	21593.4	**21294**	1.406	0.000	21726.5	21500	2.031	0.967	**21295.5**	**21294**	0.007	0
Kroe100	22068	22349.6	**22068**	1.276	0.000	22499.7	22099	1.956	0.140	**22073.4**	**22068**	0.024	0
Pr107	44303	44793.8	**44303**	1.108	0.000	44821.5	44413	1.170	0.248	**44303**	**44303**	0.000	0
Pr124	59030	59412.1	**59030**	0.647	0.000	59593.6	59030	0.955	0.000	**59030**	**59030**	0.000	0
Pr136	96772	99351.2	97547	2.665	0.801	99858.3	98499	3.189	1.785	**96775.1**	**96772**	0.003	0
Pr144	58537	58876.2	**58537**	0.579	0.000	58807.3	58574	0.462	0.063	**58537**	**58537**	0.000	0
Pr152	73682	74676.9	73921	1.350	0.324	74969.5	74172	1.747	0.665	**73695.6**	**73682**	0.018	0
Pr264	49135	50908.3	49756	3.609	1.264	52198.5	51603	6.235	5.023	**49139.5**	**49135**	0.009	0
Pr299	48191	49674.1	48310	3.078	0.247	50532.3	49242	4.858	2.181	**48194.4**	**48191**	0.007	0
Pr439	107217	115256.4	111538	7.498	4.030	116706.9	113497	8.851	5.857	**107217.6**	**107217**	0.001	0
Pr1002	259047	274419.7	270016	5.934	4.234	279419.7	273496	7.864	5.578	**259495**	**259495**	0.173	0.002

Table 4. Average ranking of MPDBA, IBA and ESA obtained by Friedman's non-parametric test for TSP.

Algorithm	Ranking
MPDBA	1,100
IBA	2,050
ESA	2,850

Table 5. The obtained unadjusted and adjusted p-values for TSP when conducting Holm's post hoc test using MPDBA as a control algorithm.

Algorithm	Unadjusted p	Adjusted p
ESA	0,000	0,025
IBA	0,000	0,050

Friedman statistic was 30.70, distributed according to a χ^2 distribution with 2 degrees of freedom. The p-value ρ for Friedman Test was 0,000. The results indicated that the Friedman statistic was significant, $\chi^2(df = 2) = 30.70$, $\rho < .01$. Thus, it is clear that there were significant differences between the tested algorithms. Notably, MPDBA has the lowest rank. Secondly, the Holm's post hoc test was conducted using MPDBA as the control algorithm to evaluate the statistical significance of the better performance of MPDBA. Table 5 shows the unadjusted and adjusted p-values obtained when applying Holm's post hoc test with MPDBA as the control algorithm. It can be concluded from the table that MPDBA is significantly better than IBA and ESA in solving TSP at a 95% confidence level.

5 Conclusion

This paper introduced a new variant of bat algorithm, called multi-population discrete bat algorithm (MPDBA), which involves three main contributions. The first one is the embedding of local search techniques, 2-opt and 3-opt, to speed up the convergence toward the optimal solution. The second one is introduction of the crossover operation. Crossover operation led to increase the intensification and, consequently, help algorithm to exploit the current search region efficiently. The third is the implementation of BA with several populations that communicate with each other from time to time. This model led to maintain the diversity and reduce the running time by evaluating many bats concurrently. The results show that MPDBA outperformed the existing algorithms, with respect to average and best solution. Furthermore, MPDBA was able to obtain the optimal solution for most instances. In future, MPDBA should be enhanced in order to get better performance to escape trapping in local optima, especially with large scale instances. Parameters such as ER, EI and their effect on MPDBA performance need further study. In addition, it would be crucial to evaluate the

performance of MPDBA to solve asymmetric TSP. Investigating MPDBA to solve other optimization problems can be studied also.

References

1. Blum, C., Roli, A.: Metaheuristics in combinatorial optimization: overview and conceptual comparison. ACM Comput. Surv. (CSUR) **35**(3), 268–308 (2003)
2. Matai, R., Singh, S.P., Mittal, M.L.: Traveling salesman problem: an overview of applications, formulations, and solution approaches. In: Traveling Salesman Problem, Theory and Applications, pp. 1–24 (2010)
3. Yang, X.-S.: A new metaheuristic bat-inspired algorithm. In: González, J.R., Pelta, D.A., Cruz, C., Terrazas, G., Krasnogor, N. (eds.) NICSO 2010. SCI, vol. 284, pp. 65–74. Springer, Heidelberg (2010)
4. Wang, G., Guo, L.: A novel hybrid bat algorithm with harmony search for global numerical optimization. J. Appl. Math. **2013**, 1–21 (2013)
5. Nguyen, T.-T., Pan, J.-S., Dao, T.-K., Kuo, M.-Y., Horng, M.-F.: Hybrid bat algorithm with artificial bee colony. In: Pan, J.-S., Snasel, V., Corchado, E.S., Abraham, A., Wang, S.-L. (eds.) Intelligent Data analysis and its Applications, Volume II. AISC, vol. 298, pp. 45–55. Springer, Heidelberg (2014). doi:10.1007/978-3-319-07773-4_5
6. Pan, T.-S., Dao, T.-K., Nguyen, T.-T., Chu, S.-C.: Hybrid particle swarm optimization with bat algorithm. In: Sun, H., Yang, C.-Y., Lin, C.-W., Pan, J.-S., Snasel, V., Abraham, A. (eds.) Genetic and Evolutionary Computing. AISC, vol. 329, pp. 37–47. Springer, Heidelberg (2015). doi:10.1007/978-3-319-12286-1_5
7. Meng, X., Gao, X., Liu, Y.: A novel hybrid bat algorithm with differential evolution strategy for constrained optimization. Int. J. Hybrid Inf. Technol. **8**(1), 383–396 (2015)
8. Khan, K., Nikov, A., Sahai, A.: A fuzzy bat clustering method for ergonomic screening of office workplaces. In: Dicheva, D., Markov, Z., Stefanova, E. (eds.) Third International Conference on Software, Services and Semantic Technologies S3T 2011. AINSC, vol. 101, pp. 59–66. Springer, Heidelberg (2011)
9. Abdel-Raouf, O., Abdel-Baset, M., El-Henawy, I.: An improved chaotic bat algorithm for solving integer programming problems. Int. J. Mod. Educ. Comput. Sci. (IJMECS) **6**(8), 18 (2014)
10. Gandomi, A.H., Yang, X.-S.: Chaotic bat algorithm. J. Comput. Sci. **5**(2), 224–232 (2014)
11. Wang, G., Guo, L., Duan, H., Liu, L., Wang, H.: A bat algorithm with mutation for UCAV path planning. Sci. World J. **2012**, 1–15 (2012)
12. Fister Jr., I., Fister, D., Yang, X.-S.: A hybrid bat algorithm. ArXiv e-prints, March 2013
13. Marichelvam, M., Prabaharan, T., Yang, X.-S., Geetha, M.: Solving hybrid flow shop scheduling problems using bat algorithm. Int. J. Logistics Econ. Globalisation **5**(1), 15–29 (2013)
14. Nakamura, R., Pereira, L., Costa, K., Rodrigues, D., Papa, J., Yang, X.-S.: BBA: a binary bat algorithm for feature selection. In: 2012 25th SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI), pp. 291–297, August 2012
15. Raghavan, S., Sarwesh, P., Marimuthu, C., Chandrasekaran, K.: Bat algorithm for scheduling workflow applications in cloud. In: 2015 International Conference on Electronic Design, Computer Networks & Automated Verification (EDCAV), pp. 139–144. IEEE (2015)

16. Sabba, S., Chikhi, S.: A discrete binary version of bat algorithm for multidimensional knapsack problem. Int. J. Bio-Inspired Comput. **6**(2), 140–152 (2014)
17. Tosun, Ö., Marichelvam, M.: Hybrid bat algorithm for flow shop scheduling problems. Int. J. Math. Oper. Res. **9**(1), 125–138 (2016)
18. Hassan, E.A., Hafez, A.I., Hassanien, A.E., Fahmy, A.A.: A discrete bat algorithm for the community detection problem. In: Onieva, E., Santos, I., Osaba, E., Quintián, H., Corchado, E. (eds.) HAIS 2015. LNCS (LNAI), vol. 9121, pp. 188–199. Springer, Heidelberg (2015). doi:10.1007/978-3-319-19644-2_16
19. Zhou, Y., Luo, Q., Xie, J., Zheng, H.: A hybrid bat algorithm with path relinking for the capacitated vehicle routing problem. In: Yang, X.-S., Bekdaş, G., Nigdeli, S.M. (eds.) Metaheuristics and Optimization in Civil Engineering. MOST, vol. 7, pp. 255–276. Springer, Heidelberg (2016). doi:10.1007/978-3-319-26245-1_12
20. Saji, Y., Riffi, M.E., Ahiod, B.: Discrete bat-inspired algorithm for travelling salesman problem. In: 2014 Second World Conference on Complex Systems (WCCS), pp. 28–31. IEEE (2014)
21. Saji, Y., Riffi, M.E.: A novel discrete bat algorithm for solving the travelling salesman problem. Neural Comput. Appl. **27**, 1–14 (2015)
22. Amara, J., Hamdani, T.M., Alimi, A.M.: A new hybrid discrete bat algorithm for traveling salesman problem using ordered crossover and 3-opt operators for bat's local search. In: 2015 15th International Conference on Intelligent Systems Design and Applications (ISDA), pp. 154–159. IEEE (2015)
23. Osaba, E., Yang, X.-S., Diaz, F., Lopez-Garcia, P., Carballedo, R.: An improved discrete bat algorithm for symmetric and asymmetric traveling salesman problems. Eng. Appl. Artif. Intell. **48**, 59–71 (2016)
24. Tsai, C.-F., Dao, T.-K., Yang, W.-J., Nguyen, T.-T., Pan, T.-S.: Parallelized bat algorithm with a communication strategy. In: Ali, M., Pan, J.-S., Chen, S.-M., Horng, M.-F. (eds.) IEA/AIE 2014. LNCS (LNAI), vol. 8481, pp. 87–95. Springer, Heidelberg (2014). doi:10.1007/978-3-319-07455-9_10
25. Heraguemi, K.E., Kamel, N., Drias, H.: Multi-population cooperative bat algorithm for association rule mining. In: Núñez, M., Nguyen, N.T., Camacho, D., Trawiński, B. (eds.) ICCCI 2015. LNCS (LNAI), vol. 9329, pp. 265–274. Springer, Heidelberg (2015). doi:10.1007/978-3-319-24069-5_25
26. Jaddi, N.S., Abdullah, S., Hamdan, A.R.: Multi-population cooperative bat algorithm-based optimization of artificial neural network model. Inf. Sci. **294**, 628–644 (2015)
27. Yang, X.-S., He, X.: Bat algorithm: literature review and applications. Int. J. Bio-Inspired Comput. **5**(3), 141–149 (2013)
28. Goldberg, D.E.: Alleles, loci, and the traveling salesman problem. In: Proceedings of an International Conference on Genetic Algorithms and Their Applications, vol. 154, pp. 154–159. Lawrence Erlbaum, Hillsdale (1985)
29. Yip, P.P., Pao, Y.-H.: Combinatorial optimization with use of guided evolutionary simulated annealing. IEEE Trans. Neural Netw. **6**(2), 290–295 (1995)

A Modified Naïve Bayes Style Possibilistic Classifier for the Diagnosis of Lymphatic Diseases

Karim Baati[1,2(✉)], Tarek M. Hamdani[1,3], Adel M. Alimi[1], and Ajith Abraham[4]

[1] REGIM-Laboratory: Research Groups on Intelligent Machines, National Engineering School of Sfax (ENIS), University of Sfax, BP 1173, 3038 Sfax, Tunisia
{karim.baati,tarek.hamdani,adel.alimi}@ieee.org
[2] Esprit School of Engineering, Tunis, Tunisia
[3] College of Science and Arts at Al-Ula, Taibah University, Al-madinah al-munawwarah, Kingdom of Saudi Arabia
[4] Machines Intelligence Research Labs (MIR Labs), Scientific Network for Innovation and Research Excellence, P.O. Box 2259, Auburn, WA 98071, USA
ajith.abraham@ieee.org

Abstract. In this paper, we propose a modified version of the Naïve Bayes Style Possibilistic Classifier (NBSPC) which has been already suggested to make decision from the categorical and subjective medical information included by the lymphography dataset of University of California Irvine (UCI). As the former NBSPC, the modified classifier combines the structure of the Naïve Bayes Classifier (NBC) as a good classifier for discrete features with the possibility theory as a powerful framework for belief estimation from subjective data. However, unlike the former NBSPC which uses the minimum as a fusion operator, the proposed classifier fuses possibilistic beliefs using the generalized minimum-based algorithm which has been recently proposed to deal with heterogeneous medical data. Experimental evaluations on the lymphograhy dataset show that the proposed G-Min-based NBSPC outperforms the former NBSPC as well as the main classification techniques which have been used in related work.

Keywords: Computer-aided diagnosis · Naïve Bayes Style Possibilistic Classifier · G-Min algorithm · Lymphatic diseases

1 Introduction

Computer-aided diagnosis (CAD) systems are systems based on interdisciplinary studies that bring together computer and medical science researchers [1] in order to assist the physician when he carries out a medical diagnosis with regard to a given patient.

When establishing CAD systems, all classification techniques, such as logic based algorithms (decision trees, learning set of rules), perceptron-based

A. Abraham et al. (eds.), *Proceedings of the 16th International Conference on Hybrid Intelligent Systems (HIS 2016)*, Advances in Intelligent Systems and Computing 552,
DOI 10.1007/978-3-319-52941-7_47

techniques (single layered perceptrons, multilayered perceptrons, Radial Basis Function (RBF) networks), statistical learning algorithms (Naïve Bayes classifiers, Bayesian networks), instance-based learning (k-Nearest Neighbour (kNN)) and Support Vector Machines (SVMs), can be called [2]. However, since decision in the medical field is extremely important, a careful choice of the classification technique is crucial in order to ensure a good performance of the CAD system.

As an example of a CAD system stemming from a careful study of the available data, we can mention the Naïve Bayes Style Possibilistic Classifier (NBSPC) which has been proposed in [3] to deal with data involved by the lymphography dataset from University of California Irvine (UCI) machine learning repository [4]. Indeed, this classifier presents the advantage to combine the structure of the Naïve Bayes Classifier (NBC) as a good classifier for discrete features [2] with the possibility theory as a powerful framework for handling subjective data [5,6].

In order to estimate possibilistic measures, the NBSPC proposed in [3] makes use of the maximum likelihood estimation as well as the probability-possibility transformation method of Dubois et al. in the discrete case [7]. Thereafter, the minimum operator is called to combine the resulting possibilistic beliefs.

Recently, a new fusion strategy based on a Generalized Minimum-based (G-Min) algorithm has been suggested as an enhancement of the minimum operator in order to fuse possibilistic beliefs stemming from heterogeneous medical data [8]. Experimental evaluations in [8] have demonstrated the efficiency of the G-Min algorithm with regard to classical fusion operators for possibilistic beliefs namely, the product and the minimum.

The main focus of the work is to hybridize the NBSPC proposed in [3] with the G-Min algorithm suggested in [8] in order to improve the classification accuracy in the diagnosis of lymphatic diseases. Obviously, that is checked through a comparison with previous studies, especially those which have evaluated Bayesian-like classifiers on the same data.

The rest of this paper is organized as follows. Section 2 illustrates main previous works dealing with the addressed diagnosis problem. Then, Sect. 3 sheds light on some basics of the possibilistic framework. Later, Sect. 4 provides details of the proposed classifier. Section 5 is devoted to show and discuss the experimental results. Finally, Sect. 6 presents some concluding remarks.

2 Related Work

In this section, only main classification techniques assessed on the lymphography dataset provided by UCI are retained. It is worth to mention, however, that all these techniques are not problem-oriented since they were evaluated on many datasets, among them we can locate the lymphography one.

A first work proposed by Clark and Niblett [9] used an induction system, CN2, and called four other algorithms namely, Assistant, the NBC, AQR and the Default Rule, for comparison. During experimentation, 70 % of the entire data was used as training samples and the remaining 30 % for testing.

Later, Madden [10] compared the performance of the Markov Blanket Bayesian Classifier (MBBC) Algorithm with three other Bayesian classifiers namely, NBC, Tree-Augmented Naïve Bayes (TAN) and a general Bayesian network. During experiments, UCI lymphography dataset was randomly divided into 2/3 for training and 1/3 for testing and the procedure was repeated 50 times to reduce variability.

In [11], Chandra and Bhaskar proposed a novel approach that rests on decision trees in order to classify patterns having categorical attributes. Experiments were conducted using a ten cross validation.

As mentioned earlier, in [3], authors evaluated a minimum-based NBSPC for the diagnosis of lymphatic diseases. During experiments, data was randomly divided, as in [10], into 2/3 for training and 1/3 for testing.

3 Possibilistic Framework

In order to recall some basics of the possibilistic framework, the following notations are considered in this section as well as in the remainder of this paper:

$C = \{c_1, c_2, ..., c_j, ..., c_n\}$: an exhaustive and exclusive universe of discourse of classes.

$A = \{a_1, a_2, ..., a_i, ..., a_m\}$: a set of attributes which may stand for either categorical or numerical values.

v_{it} : the value taken by a given attribute a_i during test.

3.1 Possibility Theory

Possibility theory, introduced by Zadeh [12] and then developed by Dubois and Prade [13] is a fusion theory based on fuzzy sets theory and devoted to represent and combine imperfect information in a qualitative or quantitative way. Information imperfections treated by possibility theory may stand for the uncertainty due to variability of observations, the uncertainty due to poor information, the information imprecision, the information ambiguity, etc. [15]. Furthermore, it has been shown that possibility theory is well convenient, in particular, to treat subjective information [5,6].

At the semantic level, the basic function in possibility theory is a possibility distribution denoted as π which assigns to each possible class c_j from C a value in [0, 1]. The possibility value assigned to a class c_j stands for plausibility i.e. the belief degree that this class is the right one.

By convention, $\pi(c_j) = 1$ means that c_j is totally possible and if $\pi(c_j) = 0$, c_j is considered as impossible. Intermediary values distinguish values which are more possible than others. Finally, note that in the normalization version of a possibility distribution, we require that at least one class of C is totally possible.

From the possibility distribution, two dual measures, namely possibility Π and necessity N, may be assigned to each subset E of the power set 2^C as follows:

$$\begin{aligned} \Pi(E) &= max[\pi(c_j)], c_j \in E \\ N(E) &= min[1 - \pi(c_j)], c_j \notin E \end{aligned} \tag{1}$$

Between these two measures there is a duality relation defined by :

$$N(E) = 1 - \Pi(\bar{E}), E \subset 2^C \tag{2}$$

$\Pi(E) = 1$ corresponds to the situation where at least one singleton $c_j \in E$ has a possibility degree equal to 1 and it is likely to be the right class but without being necessary. $N(E) = 1$ corresponds to the situation where $\Pi(\bar{E}) = 0$, meaning that the right class certainly does not belong to subset $\bar{E}$, so it is certainly a member of the subset $E \subset 2^C$.

On the other hand, as being based on fuzzy sets theory, possibility theory may recall all the panoply of fuzzy combination rules in order to fuse possibility estimates of a given class.

3.2 Conditional Possibility

Possibilistic conditioning corresponds to revising an initial possibility distribution when a new information becomes available [16]. In possibility theory, conditioning may be defined through a counterpart of the Bayes rule. Indeed, it can be stated for two subsets E and F of 2^C by:

$$\Pi(E \cap F) = \Pi(E|F) * \Pi(F) \tag{3}$$

where * is a combination operator which is commonly chosen as the minimum or the product [13].

3.3 Naïve Bayes Style Possibilistic Classification

Similarly to Bayesian classification, possibilistic classification is based on the aforementioned possibilistic version of the Bayes rule and can be defined by:

$$\begin{aligned} &\pi(c_j|a_1 = v_{1t}, ..., a_m = v_{mt}) \\ &= \frac{\pi(a_1 = v_{1t}, ..., a_m = v_{mt}|c_j) * \pi(c_j)}{\pi(a_1 = v_{1t}, ..., a_m = v_{mt})} \end{aligned} \tag{4}$$

By assuming that there is no a priori knowledge about classes and the input vector to be classified, we can take $\pi(c_j) = 1$ and $\pi(a_1 = v_{1t}, ..., a_m = v_{mt}) = 1$. Further, as in naïve Bayesian combination, naïve possibilistic classification assumes that attributes $\{a_i\}$ $(\forall \ \ 1 \leq i \leq m)$ are independent. The conditional joint possibility $\pi(a_1 = v_{1t}, ..., a_m = v_{mt}|c_j)$ is equal in that case to the fusion of conditional possibility estimations stemming from each single attribute a_i. Therefore, in a context characterized by no a priori knowledge about classes and input vector as well as independent attributes, Eq. 4 becomes [14] :

$$\begin{aligned} &\pi(c_j|a_1 = v_{1t}, ..., a_m = v_{mt}) \\ &= \pi(a_1 = v_{1t}|c_j) * ... * \pi(a_m = v_{mt}|c_j) \end{aligned} \tag{5}$$

As stated for the conditioning rule, * may be taken as either the product or the minimum [17,18].

In practice, given a new instantiation $\{a_1 = v_{1t}, ..., a_m = v_{mt}\}$, we must establish a matrix Π of possibilistic estimates in order to perform the product-based or the minimum-based classification. This matrix is defined as follows :

$$\Pi = \{\pi(i,j)\} = \{\pi(a_i = v_{it}|c_j)\} \tag{6}$$

$\forall \;\; 1 \leq i \leq m$ and $\forall \;\; 1 \leq j \leq n$

Lastly, the final decision stands for the class c^* for which Eq. 5 yields the highest degree of possibility belief:

$$c^* = arg \quad \max_j(\pi(1,j) * ... * \pi(m,j)) \tag{7}$$

4 Proposed Classifier

In order to perform the classification of the categorical and subjective data of the lymphography dataset, the proposed classifier rests on two main steps, namely the estimation of possibility beliefs and then the fusion of these beliefs using the G-Min algorithm. In the following, technical details of these two steps are detailed.

4.1 Estimation of Possibility Beliefs

To illustrate the method used within the estimation step, let consider $V_i = \{v_{i1}, v_{i2}, ..., v_{is}, ..., v_{iz}\}$ the set of possible categorical values for a given attribute a_i. In order to build possibility distributions from V_i, we start by computing the conditional probability measures $\{p(a_i = v_{is}|c_j)\}$ $(\forall \;\; 1 \leq s \leq z)$ for each attribute a_i with respect to each class c_j using the maximum likelihood estimation as follows:

$$p(a_i = v_{is}|c_j) = \frac{\#(a_i = v_{is}, c_j)}{\#c_j} \tag{8}$$

where $\#(a_i = v_{is}, c_j)$ is the number of training samples belonging to the class c_j and having the value v_{is} for the attribute a_i and $\#c_j$ is the number of training samples that belong to the class c_j

Afterward, we use the probability-possibility transformation of Dubois et al. in the discrete case [7] in order to express possibilistic estimates [3]. This transformation is defined by :

$$\pi(a_i = v_{is}|c_j) = \sum_{v_{ir}|p(v_{ir}) \leq p(v_{is})} p(a_i = v_{ir}|c_j) \tag{9}$$

with $1 \leq r \leq z$

Example: Let consider a categorical attribute a_2 with three possible values and let assume that $p(a_2 = v_{21}|c_3) = 0.7$, $p(a_2 = v_{22}|c_3) = 0.2$ and $p(a_2 = v_{23}|c_3) = 0.1$. The corresponding possibilistic estimates are: $\pi(a_2 = v_{21}|c_3) = 0.7 + 0.2 + 0.1 = 1$, $\pi(a_2 = v_{22}|c_3) = 0.2 + 0.1 = 0.3$ and $\pi(a_2 = v_{23}|c_3) = 0.1$

4.2 Fusion of Possibilistic Beliefs

To fuse possibilistic beliefs, the minimum rule allocates the final decision to the class which satisfies:

$$c^* = arg \quad \max_j(\min_i \pi(i,j)) \tag{10}$$

where $\Pi = \{\pi(i,j)\}$ $(1 \leq i \leq m$ and $1 \leq j \leq n)$ the matrix of possibilistic estimates which has been defined in Eq. 6.

In many cases, the minimum-based possibilistic combination is likely to lead to a final decision that may have very close possibility estimate to other alternatives. In such situation, the quality of decision may be seriously altered since the final class is likely to be inaccurate. In order to resolve this problem, the G-Min algorithm has been proposed in [8] in order to avoid the ambiguity between the final decision and the rest of classes and hence to find a decision with a possibility estimate widely away from other alternatives.

The G-Min algorithm requires the matrix Π of possibilistic estimates and it is based on two main steps. The first aims to build a set of possible decisions whereas the second aims to filter this set in order to find a final class with a high score of reliability [8].

In the first step, columns of the matrix of possibilistic estimations Π are sorted increasingly and links with attributes are left out. Then, rows of the resulting matrix are sorted decreasingly and the argument (the class) related to each value is retained. Afterward, a decision set *dset* is established using the first row of the output matrix Π after the two sorting steps [8].

In the second step, the next row of Π is treated by picking possibility values of respectively each of the selected classes on *dset*. Based on the new possibility values, we eliminate each class of the decision set *dset* that have not a possibility value within the threshold α with respect to the maximum of the updated possibility values. Each time we find *dset* different from a singleton, we deal with the next row of Π in order to try to further filter the decision set. The second step is repeated until the decision set *dset* is a singleton or we are in the last row of the matrix Π. The final class is then assigned to the one having the maximum value of possibility in the decision set *dset* [8].

More details about the proposed algorithm may be found in Algorithm 1 [8].

In order to explain more the proposed algorithm, we exhibit a numerical example in Fig. 1 in which we have taken $\alpha = 0.1$. In this example, cardinality of the decision set has decreased along rows and a unique class has been obtained at the end. One could plainly see that the final decision in this example differs from the one that would be chosen if the minimum operator has been employed.

Algorithm 1. G-Min Algorithm

Require: Matrix of possibilistic estimations Π
Ensure: Final decision c^*
  $\Pi \leftarrow sort(\Pi,'rows','increase')$
  $\Pi \leftarrow sort(\Pi,'columns','decrease')$
  $q \leftarrow 1$
  $dset[q] \leftarrow arg(\pi(1,1))$
  $c^* \leftarrow arg(\pi(1,1))$
  for $j \leftarrow 2$ *to* n **do**
    if $\pi(1,1) - \pi(1,j) < \alpha$ **then**
      $q \leftarrow q + 1$
      $dset[q] \leftarrow arg(\pi(1,j))$
    end if
  end for
  $card \leftarrow cardinality(dset)$
  $i \leftarrow 2$
  while $card \neq 1$ *and* $i \neq m$ **do**
    for $q \leftarrow 1$ *to* $dset.size$ **do**
      $poss[q] \leftarrow arg^{-1}(dset[q], i)$
    end for
    $max \leftarrow maximum(poss)$
    for $q \leftarrow 1$ *to* $poss.size$ **do**
      if $max - poss[q] \geq \alpha$ **then**
        $dset \leftarrow dset.remove(arg(poss[q])$
        $c^* \leftarrow max$
      end if
    end for
    $card \leftarrow cardinality(dset)$
    $i \leftarrow i + 1$
  end while

5 Experimental Evaluations

In order to assess the performance of the G-Min-based NBSPC to diagnose lymphatic diseases, experiments are conducted, as mentioned earlier, on the lymphography dataset provided by UCI.

UCI lymphography dataset was obtained from the University Medical Centre, Institute of Oncology, Ljubljana, Yugoslavia. It contains 148 instances in total and there are 18 categorical valued attributes which are gathered from radiologists judgments and estimations (Table 1). Furthermore, there are four classes in the class attribute namely, normal, metastases, malign lymph and fibrosis.

To apply the proposed classifier, we have called the experimental methodology used in [3,10]. Indeed, the dataset was randomly divided into 2/3 for training and 1/3 for testing and the procedure was repeated 50 times to reduce variability. Further, we have taken the threshold $\alpha = 0.1$ when running the G-Min algorithm.

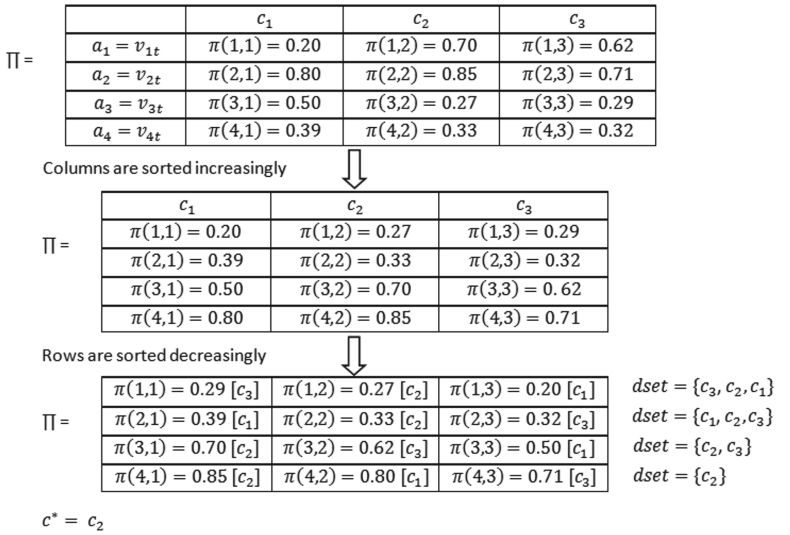


Fig. 1. A numerical example for the application of the G-Min algorithm

Table 1. UCI lymphography dataset attributes

Attribute	Possible values
Lymphatics	1 : normal; 2 : arched; 3 : deformed; 4 : displaced
Block of afferent	1: no; 2: yes
Block of lymph c	1: no; 2: yes
Block of lymph s	1: no; 2: yes
By pass	1: no; 2: yes
Extravasates	1: no; 2: yes
Regeneration	1: no; 2: yes
Early uptake	1: no; 2: yes
Lymph nodes dimension	ranges from 0 to 3
Lymph nodes enlarge	ranges from 1 to 4
Changes in lymph	1: bean; 2: oval; 3: round
Defect in node	1: no; 2: lacunars; 3: lacunars marginal; 4: lacunars central
Changes in node	1: no; 2: lacunars; 3: lacunars marginal; 4: lacunars central
Changes in structure	1: no; 2: grainy; 3: drop-like; 4: coarse; 5: diluted
	6: reticular; 7: strippped; 8: faint
Special forms	1: no; 2: chalices; 3: vesicles
Dislocation of node	1: no; 2: yes
Exclusion of node	1: no; 2: yes
Number of nodes	ranges from 0 to 80

Table 2. Experimental results obtained with various classifiers applied on the UCI lymphography dataset

Work	Classification technique	Accuracy (%)
Clark and Niblett [9]	CN2	76
	Assistant	79
	NBC	83
	AQR	82
	Default rule	56
Madden [10]	MBBC	83.47 ± 9.45
	NBC	83.60 ± 9.82
	TAN	85.47 ± 9.49
	Bayesian network	81.47 ± 10.4
Chandra and Bhaskar [11]	Decision Tree using gain ratio	72.86
Clarck and Niblett [9]	Decision Tree using a novel split measure	77.64
Baati et al. [3]	Minimum-based NBSPC	86.27 ± 9.92
Our work	G-Min-based NBSPC	**87.76 ± 9.60**

Table 2 shows that the proposed classifier achieves an average classification accuracy of 87.76 % and hence outperforms the former NBSPC as well as all Bayesian classifiers which have been similarly evaluated on the same dataset (namely, MBBC Algorithm, NBC, TAN and Bayesian network). Moreover, even not similarly tested, our classifier shows better accuracy than all classification techniques proposed in [9,11], respectively.

6 Conclusion

This paper has proposed a CAD system for the diagnosis of lymphatic diseases from data involved by the UCI lymphography dataset. The proposed CAD system is based on a classifier that hybridizes the capability of the NBSPC to estimate beliefs from categorical and subjective data with the efficiency of the G-Min as a novel algorithm for the fusion of possibilistic beliefs. Experimental results have shown that the proposed classifier outperforms the former NBSPC as well as the other classification techniques which have been assessed on the same data.

The good performance of the proposed CAD system is appealing for two main reasons. First, the system may be kept and reinforced by a feature selection stage in order to establish a more accurate CAD for the addressed medical problem. Second, possibility theory in general and the proposed classifier in particular, may be useful for other CAD systems dealing with medical features that share the same specifications with those treated in the current study.

Aknowledgment. The authors would like to acknowledge the financial support of this work by grants from General Direction of Scientific Research (DGRST), Tunisia, under the ARUB program.

References

1. Jemmaa, A.B., Ltifi, H., Ayed, M.B.: Multi-agent architecture for visual intelligent remote healthcare monitoring system. In: Abraham, A., Han, S.Y., Al-Sharhan, S.A., Liu, H. (eds.) HIS 2015. AISC, vol. 420, pp. 211–221. Springer, Heidelberg (2016). doi:10.1007/978-3-319-27221-4_18
2. Kotsiantis, S.B.: Supervised machine learning: a review of classification techniques. Informatica **31**, 249–268 (2007)
3. Baati, K., Hamdani, T.M., Alimi, A.M.: Diagnosis of lymphatic diseases using a naïve bayes style possibilistic classifier. In: Proceedings of the IEEE International Conference on Systems, Man and Cybernetics (SMC), pp. 4539–4542. IEEE (2013)
4. http://archive.ics.uci.edu/ml/datasets/Lymphography
5. Raufaste, E., Neves, R.D.S., Marin, C.: Testing the descriptive validity of possibility theory in human judgments of uncertainty. Artif. Intell. **148**(1), 197–218 (2003)
6. Alsun, M.H., Lecornu, L., Solaiman, B., Le Guillou, C., Cauvin, J.M.: Medical diagnosis by possibilistic classification reasoning. In: 13th Conference on Information Fusion (FUSION), pp. 1–7. IEEE, July 2010
7. Dubois, D., Foulloy, L., Mauris, G., Prade, H.: Probability-possibility transformations, triangular fuzzy sets, and probabilistic inequalities. Reliab. Comput. **10**(4), 273–297 (2004)
8. Baati, K., Hamdani, T.M., Alimi, A.M., Abraham, A.: A new possibilistic classifier for heart disease detection from heterogeneous medical data. Int. J. Comput. Sci. Inf. Secur. **14**(7), 443–450 (2016)
9. Clark, P., Niblett, T.: Induction in noisy domains. In: Progress in Machine Learning, pp. 11–30 (1987)
10. Madden, M.G.: Evaluation of the performance of the Markov blanket Bayesian classifier algorithm. arXiv preprint cs/0211003 (2002)
11. Chandra, B., Bhaskar, S.: A new approach for classification of patterns having categorical attributes. In: Proceedings of the IEEE International Conference on Systems, Man and Cybernetics (SMC), pp. 960–964. IEEE (2011)
12. Zadeh, L.A.: Fuzzy sets as a basis for a theory of possibility. Fuzzy Sets Syst. **1**(1), 3–28 (1978)
13. Dubois, D., Prade, H.M., Farreny, H., Martin-Clouaire, R., Testemale, C.: Possibility Theory: An Approach to Computerized Processing of Uncertainty, vol. 2. Plenum press, New York (1988)
14. Borgelt, C., Gebhardt, J.: A naïve bayes style possibilistic classifier. In: Proceedings of the 7th European Congress on Intelligent Techniques and Soft Computing (1999)
15. Khaleghi, B., Khamis, A., Karray, F.O., Razavi, S.N.: Multisensor data fusion: a review of the state-of-the-art. Inf. Fusion **14**(1), 28–44 (2013)
16. Dubois, D., Prade, H.: The logical view of conditioning and its application to possibility and evidence theories. Int. J. Approx. Reason. **4**(1), 23–46 (1990)
17. Baati, K., Hamdani, T.M., Alimi, A.M.: Hybrid naïve possibilistic classifier for heart disease detection from heterogeneous medical data. In: Proceedings of the 13th International Conference on Hybrid Intelligent Systems, pp. 235–240. IEEE (2013)
18. Baati, K., Hamdani, T.M., Alimi, A.M.: A modified hybrid naïve possibilistic classifier for heart disease detection from heterogeneous medical data. In: Proceedings of the 6th International Conference on Soft Computing and Pattern Recognition, pp. 353–358. IEEE (2014)

A Scheduling Algorithm for Beacon Message in Vehicular Ad Hoc Networks

Heithem Nacer[1(✉)] and Mohamed Mazouzi[2]

[1] Department of Informatics, Faculty of Siences of Gabes, University of Gabes, Gabes, Tunisia
nacerheithem@gmail.com
[2] Higher Institute of Business Administration of Sfax, Member of CES-Laboratory, University of Sfax, Sfax, Tunisia
mohammad.mazouzi@yahoo.fr

Abstract. Vehicular Ad hoc NETwork (VANET) was proposed in order to prevent accidents and to improve road safety. Indeed, IEEE 1609.4 was developed to support multi-channel mechanism to provide both safety and non-safety applications. The CCH interval is also a key parameter for the 802.11p MAC protocol. In order to get a wide view of the different techniques used to broadcast a message, we evaluate the performance of the 802.11p MAC protocol with various vehicle densities and different CCH interval settings. Moreover, we propose SABM, a Scheduling Algorithm for vehicles attempting to transmit a Beacon Message, which firstly adjusts the CCH interval according to the road traffic and then schedule the safety messages based their priorities. The simulation results show that SABM outperforms the IEEE 802.11p MAC protocol. On one hand, we can significantly reduce the delivery delay and the collision probability, on the other hand, at the same time equilibrating the channel utilization ratio during CCH interval.

Keywords: Vehicular ad-hoc networks · IEEE 802.11p · MAC · Beacon messages · Broadcast · Collision · Delay · Throughput

1 Introduction

Vehicular Ad Hoc Networks (VANETs) are considered as a special case of mobile ad hoc networks (MANET) [1]. The IEEE 1609.4 protocol was presented to improve the dissemination of messages in VANETs by adding the concept of multi-channel standard IEEE 802.11p. These networks provide many types of applications, such comfort applications and road safety applications to avoid traffic jams and reduce time spent on roads. The DSRC standard, Dedicated Short Range Communication [2], was specifically designed for communications in vehicular networks by reserving specific radio frequencies to these networks [3]. The Federal Communications Commission (FCC), responsible for the allocation of frequency bands in the US, has assigned a bandwidth of 75 MHz spectrum in the 5.850–5.925 GHz. DSRC presents seven different channels; each one is of 10 MHz, as shown in Fig. 1. These seven channels include a control channel CCH (Ch.178) and six service channels SCH (Ch.172, Ch.174, Ch.176, Ch.180, Ch.182 and Ch.184).

A. Abraham et al. (eds.), *Proceedings of the 16th International Conference on Hybrid Intelligent Systems (HIS 2016)*, Advances in Intelligent Systems and Computing 552,
DOI 10.1007/978-3-319-52941-7_48

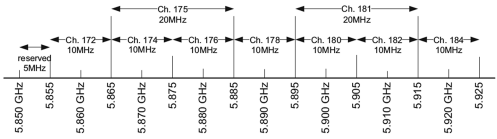


Fig. 1. The channel assignment of DSRC [4]

It divides the synchronization interval, with duration of 100 ms, into two equal times of 50 ms. The first one is for sending safety messages on the CCH channel in order to maximize the receipt of these priority messages. During the second interval, the vehicles are free to choose their listening channel. To allow the switch of listening channel, a guard interval of 4 ms is triggered. During the guard time, the channel is considered busy, and no vehicle can transmit message. The CCH interval is used for periodic broadcast of control information. However, in the high density of vehicles, the limited length of the CCH may be unable to provide sufficient channel capacity to provide a wide range of road safety messages. Indeed, if the node density is low, the CCH resource will be wasted. We shall focus on safety messages because they have strong constraints in terms of messages delivery time and quality of service.

The rest of this article is organized as follows. In the second section, related works are discussed. After that, IEEE 802.11p MAC is reassessed in the third section. Then, in the fourth section our proposed algorithm SABM for the MAC 802.11p will be detailed. Finally, the conclusions and possibilities for future studies are provided in the last section.

2 Related Works

Dedicated Multichannel MAC protocol (DMMAC) has been proposed in [6] to perform a variable length of the CCH interval based on adaptive broadcast mechanism for a safety message transmission without collision and limited delay. However, the dynamically adjusting of the synchronization interval has not been considered.

Based on [8], Variable CCH Interval (VCI), a multi-channel MAC mechanism is proposed that divides the CCH interval in two, one for safety messages and the other for the warning service (WAVE Service announcement). Depending on network conditions, this mechanism can dynamically adjust the relation between CHC and SCHs intervals. Although the VCI mechanism is able to provide an efficient use of CCH and SCHs channels to some extent, it allows working well only in limited scenarios with low utilization channel.

The authors in [10] proposed a MAC protocol (VER-MAC) that enables the nodes to broadcast safety messages twice during both CCH and SCH intervals which increases the reliability of the secure broadcast. However the VER-MAC average delay is greater than that of the IEEE 1609.4 since it requires the addition of complex data structures, thus it suffers from further delay of emergency packets.

In [11], VEMMAC (Vehicular Enhanced Multichannel MAC) has been proposed. It adopts the IEEE 1609.4 standard with sequences of CCH intervals and alternative

SCH. VEMMAC allows nodes to transmit non-safety messages during the CCH interval and dissemination of safety messages twice with each CCH and SCH interval. However the system is unable to monitor the high collision in the beginning of CCH and that of SCH intervals. Therefore, the nodes could lose emergency packets on the CCH interval due to the extended transmission mode.

3 Performance Evaluation of IEEE 802.11p MAC Protocol

This section presents the performance evaluation of the IEEE 802.11p MAC [13] protocol in safety applications. Its evaluation is based on OMNET++ [14] for network simulation framework and Veins [15] as the core of vehicular simulation framework, which extends the network simulator to cover vehicular communication. For the road traffic simulator and providing realistic node mobility, SUMO (Simulation of Urban Mobility) [16] is integrated to OMNET++ and Veins framework. To evaluate our work performance, we vary the traffic load on the channel, so we change the number of vehicles sharing the same network range and introduce the mobility.

As shown in Fig. 2, the generated traffic scenario used in our performance evaluation is presented. The realistic highway map has been imported from Open Street Map (OSM) [17] covering 1000 m highway. Each direction of the highway consists of two lanes. The scenario includes a number of vehicles varying from 20 to 200 where they are moving along a 1000 m long road with maximum speed of 120 km/h. In this scenario, each vehicle has a maximum communication range of 1000 m and disseminates messages with 39 bytes packet size. Every 100 ms, each vehicle sends one status safety message called Beacon. The Beacon messages are generated randomly during the CCH interval called (CCHI). The CCHI is set respectively as [10, 20, 30, 40, 50, 60, 70, 80, 90, 100] in order to evaluate the performance of 802.11p MAC at various CCH intervals. Table 1 shows the value parameters that are used in the simulation scenarios.

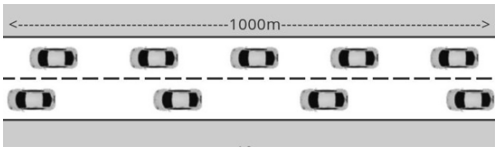


Fig. 2. Simulation test scenario

The following three metrics are used to evaluate the 802.11p MAC performance.

- Average End-to-End Beacon Delay: measures the average duration taken by a message to travel from the source node to destination node.
- Throughput of Beacon: measures the average number of successfully delivered packets.
- Collision probability: measures the average collision probability can be occurred.

Table 1. Parameters settings.

Parameter	Value
Size of beacon message Lbeacon	39 bytes
Size of MAC header Lmac	32 bytes
Data rate (Rate)	6 Mbps
Communication range	1000 m
Maximum transmission power	760 mw
Receiver sensitivity	-82dBm
Simulation time	20 s

Figure 3 shows the average beacon delay for different vehicle density and CCH interval settings. In general, we note that the beacon delay in low density is higher than in high density so it can be seen that the delay increases according to the increase in vehicle density.

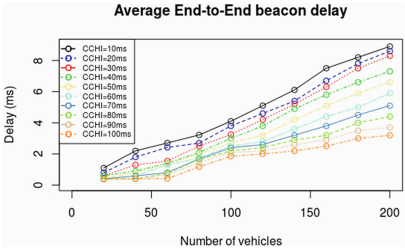


Fig. 3. Average End-to-End Beacon delay

As shown in Fig. 4, system throughput performance comparison of the different scenarios with various numbers of vehicles and CCH settings is evaluated. In this scenario, it can be noticed that for safety application the beaconing throughput increases according to the increase in number of vehicles, this is due to large number of nodes in the network then large number of beacons will be broadcasted.

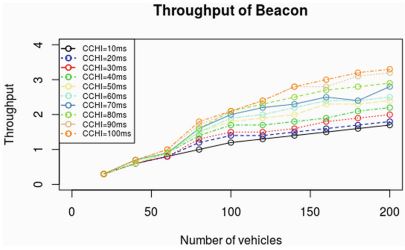


Fig. 4. Throughput of Beacon

Figure 5 presents the average collision probability obtained from the different scenarios. It can be seen that the beacon loss probability due to collisions increases with

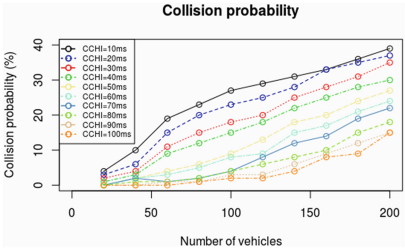


Fig. 5. Collision probability

the traffic load increase and CCH interval decrease. Moreover, the scenario with higher density and shorter CCH interval suffer from higher lost messages than the scenario with lower number of vehicles and longer CCH interval.

To conclude, in VANET safety applications, the performance of the IEEE 802.11p MAC protocol can be improved significantly in terms of collision, delay and throughput first by adjusting the CCH interval according to the vehicle density and then by scheduling the beacon message.

4 SABM: A Scheduling Algorithm for Beacon Message in Vehicular Ad Hoc Networks

In this section, we propose an enhanced scheduling algorithm for beacon message during CCH interval (*SABM*). At the first we adjust the CCH interval according to the road traffic. Then, we schedule safety messages by their priority for accessing the wireless channel. *SABM* aims firstly to minimize the collision probability and the delivery delay. Secondly, it maximizes the number of vehicles that receive the message. Thirdly, it equilibrates the channel utilization ratio during CCH interval.

For that reason, *SABM* contains three steps: CCH interval adjusting, CCH interval dividing to sub-intervals, Priority-Based message scheduling.

4.1 CCH Interval Adjusting

This algorithm contains two sub-steps: Vehicle counting and CCH interval calculating.

Vehicle counting: In this work, our first step is to adapt the CCH interval in accordance with the average number of vehicles in a highway. So this algorithm aims to calculate the number of vehicles registered by each Road Site Unit (RSU). The detail of the vehicle counting algorithm is presented in Algorithm 1. As shown in this algorithm, N_i indicates the number of vehicles registered by RSU_i. Furthermore, to simplify the algorithm the collision is supposed to result only between two vehicles. As presented in Fig. 6, the N_i will be sent to the control center to calculate the new duration of CCH interval in accordance with the average number of vehicles in a highway.

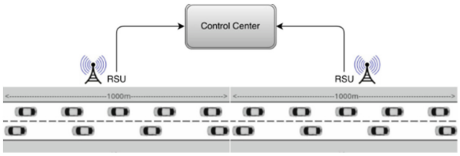


Fig. 6. CCH interval adjusting example

Algorithm 1. Vehicle counting algorithm:

```
Ni←Numbers of contention vehicles 0
if received beacon message successfully then
Ni = Ni+1
else if (failed to receive a clear message without collision) then
Ni = Ni+2
end if
```

CCH interval calculating: The previous step has given us N_i, the number of vehicles registered by RSU_i. All N_i will be collected periodically by the control center, responsible for calculating the CCH interval. Suppose a route segment contains j RSUs; the average number of contention vehicles N_{ave} can then be determined as:

$$N_{ave} = \frac{\sum_{1}^{j} N_i}{j} \tag{1}$$

To determine the duration T_{cch} of the CCH interval, we will use N_{max} the maximum number of existing vehicles in a well-defined road segment. The Eq. 2 is used to calculate the CCH interval length T_{cch}.

$$T_{cch} = \alpha \times SI \tag{2}$$

Where $\alpha = \frac{N_{ave}}{N_{max}}$ and *SI* is the synchronization interval 100 ms.
The details of CCH interval calculating algorithm are shown in Algorithm 2.

Algorithm 2. CCH interval calculating algorithm:

```
N←0
Nave←0
j←Numbers of RSUs
for (i=1; i<=j;i++) do
  receive Ni from RSUi
N = N+Ni
end for
Nave=N/j
Nmax←Maximum number of existing vehicles in a highway segment
 = Nave/Nmax
SI←100ms, the length of the synchronization cycle
Tcch= α ✕ SI
Broadcast Tcch to all RSUs
```

4.2 CCH Interval Dividing

After calculating the CCH interval and in order to use a scheduler in the MAC layer level to decide what message to transmit the first, we divide the CCH interval with its new value T_{cch} to N_{ave} periods, each one is of length ($t = MessageLength/Rate$). During this period, a vehicle can send its message. We assume $T = \{T_1 \ldots T_{Nave}\}$ a set of period's time for sending a message.

4.3 Message Priority-Based Scheduling

After the division of CCH interval to sub periods and to determine which message to transmit the first, safety messages will be scheduled based on the *PriorityQueueing* parameter as defined in the below equation:

$$PriorityQueueing = MessageArrival + MaxLatency \tag{3}$$

As discussed previously, we have N_{ave} messages to be transmitted during the CCH interval and a set of periods $T = \{T_1 \ldots T_{Nave}\}$ for sending a message. Therefore, the message with the lower value of *PriorityQueueing* was the most critical content and must be transmitted first so this message will have the period of time T_1 contrary to the message of great value *PriorityQueueing*, which occupies the period T_{Nave}.

5 Simulation Results and Discussions

The proposed algorithm is evaluated utilizing the same simulation configuration as discussed in Sect. 3.

5.1 Average End-to-End Beacon Delay

In Fig. 7 the average End-to-End beacon message delay for different vehicles density is presented. The results showed the better performance of proposed solution SABM in terms of delay. In dense vehicular environments SABM is good compared to the 802.11p. When the vehicles density is less than the threshold (30), the delay of the proposed solution is slightly longer than the 802.11p and it is slightly shorter when the vehicles density is greater than the threshold (30).

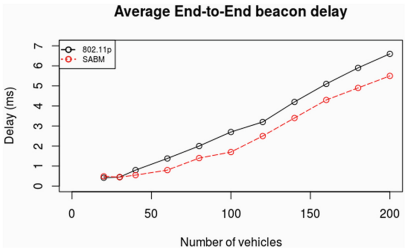


Fig. 7. Average End-to-End Beacon delay

5.2 Throughput of Beacon

As shown in Fig. 8, the beaconing throughput rapidly increases with the increase in the number of nodes since more number of beacons will be broadcasted in a network. Moreover, the throughput is strongly related to the collision problem but in our proposed scheme this problem is relatively resolved using the scheduling mechanism so we have less collision compared to 802.11p then more successfully reception messages as a result a higher value of beaconing throughput.

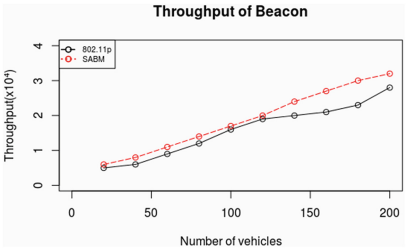


Fig. 8. Throughput of Beacon

5.3 The Collision Probability

As presented in Fig. 9, it is shown the better performance of our proposed scheduling mechanism SABM in the terms of collision probability. When the number of nodes is greater than the threshold (30) obtained in this simulation scenario, our proposed approach is good compared to 802.11p scheme. It decreases the number of collisions therefore increase the reception rate.

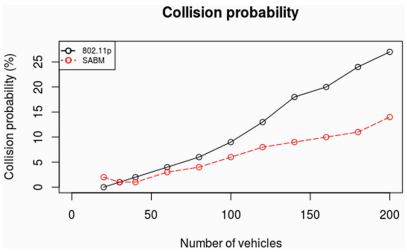


Fig. 9. Collision probability

6 Conclusions and Future Work

In this paper, we proposed a Scheduling Algorithm for Beacon Message in Vehicular Ad Hoc Networks during CCH interval, called SABM. It adjusts CCH interval according to the road traffic. Then, it schedules safety messages based on their priorities for accessing the wireless channel. We have testified it and the results of our simulation show that it better outperforms the original one defined in IEEE 802.11p.

In this work, we focused on safety messages during CCH interval and for the future work; we will implement and evaluate an algorithm to IEEE 802.11p standard for non-safety messages during SCH interval.

References

1. Marias, G.F., Georgiadis, P., Flitzanis, D., Mandalas, K.: Cooperation enforcement schemes for manets: a survey. Wirel. Commun. Mob. Comput. **6**(3), 319332 (2006)
2. ASTM International, Standard Specification for Telecommunications and Information Exchange Between Roadside and Vehicle Systems 5 GHz Band Dedicated Short Range Communications (DSRC) Medium Access Control (MAC) and Physical Layer (PHY) Specifications, April 2009
3. Chia, Y.S., Siew, Z.W., Kiring, A., Yang, S.S., Teo, K.T.K.: Adaptive hybrid channel assignment in wireless mobile network via genetic algorithm. In: International Conference on Hybrid Intelligent Systems (HIS) (2011)
4. IEEE: IEEE guide for wireless access in vehicular environments (WAVE) Architecture. IEEE Std 1609.0-2013, p. 178 (2014)
5. Qi, C., Jiang, D., Delgrossi, L.: IEEE 1609.4 DSRC multichannel operations and its implications on vehicle safety communications. In: Proceedings of the IEEE Vehicular Networking Conference, pp. 1–8 (2009)
6. Liu, N., Ji, Y.S., Liu, F.Q., Wang, X.H.: A dedicated multichannel MAC protocol design for VANET with adaptive broadcasting. In: Proceedings of the IEEE Wireless Communications and Networking Conference, pp. 1–6 (2010)
7. Wang, Q., Leng, S., et al.: An IEEE 802.11p-based multi-channel MAC scheme with channel coordination for vehicular Ad-Hoc networks. IEEE Trans. Intell. Trans. Syst. **7**(2), 449–458 (2012)
8. Wang, S.Y., Chou, C.L., Liu, K.C., Ho, T.W., Hung, W.J., Huang, C.F., Hsu, M.S., Chen, H.Y., Lin, C.C.: Improving the channel utilization of IEEE 802.11p/1609 networks. In: Proceedings of the Wireless Communications and Networking Conference (WCNC 2009), pages 16 (2009)
9. Zarei, M., Rahmani, A.M., Farazkish, R., Zahirnia, S.: Fairness congestion control for a disTrustful wireless sensor network using Fuzzy logic. In: International Conference on Hybrid Intelligent Systems (HIS) (2010)
10. Dang, D., Hong, C., Lee, S., Huh, E.: An efficient and reliable MAC in VANETs. IEEE Commun. Lett. **18**(4), 616619 (2014)
11. Dang, D.N.M., Dang, H.N., Do, C.T., Hong, C.S.: An enhanced multichannel MAC for vehicular ad hoc networks. In: IEEE Wireless Communications and Networking Conference (WCNC), pp. 351–355 (2013)
12. Hafeez, K.A., Zhao, L., Mark, J.W., Shen, X., Niu, Z.: Distributed multichannel and mobility-aware cluster-based MAC protocol for vehicular ad hoc networks. IEEE Trans. Veh. Technol. **62**(8), 38863902 (2013)
13. Becker, P., Gotzhein, R., Kuhn, T.: MacZ - A Quality-of-Service MAC Layer for Ad-hoc Networks. In: International Conference on Hybrid Intelligent Systems (HIS) (2007)
14. Omnetpp.org. OMNeT++ discrete event simulator Home http://www.omnetpp.org
15. Veins.car2x.org. Veins. http://veins.car2x.org. Accessed June 2016
16. Pattberg, B.: DLRInstitute of transportation systemsSUMO Simulation of urban mobility. Sumo-sim.org. http://sumo-sim.org/. Accessed June 2016
17. Openstreetmap.org. OpenStreetMap. http://www.openstreetmap.org/

Iris Localization Using Mixture of Probability Distributions in the Segmentation Process

Fatma Mallouli(✉) and Mohamed Abid

Laboratory CES-Lab National School of Engineering, Sfax, Tunisia
fatmamallouli@yahoo.com

Abstract. Iris segmentation plays an important role in an accurate iris recognition system. In this paper, we contribute to a more accurate iris segmentation process, hence, a more precise identification tool. We introduce a novel iris segmentation approach based on mixture of probability distributions modeling and an extended Expectation Maximization based on the (EM) algorithm. A segmentation approach is presented by exploring different techniques. Our analysis takes care of segmenting the eye image using the Markovian case and the independent case. In each case, the approach uses different mixture of probability distributions. While providing comparable results, our update algorithm gives better segmentation results based on our learning system. The accuracy of our algorithm is proved by Kulback-Leibler divergence and mean squared error computations. The statistical significance of our results are evaluated using Anova Test. The proposed method was applied on the CASIA testing database by Chinese Academy of Sciences Institute of Automation-Iris-Twins. The results showed, for both cases, the accuracy of the proposed algorithm verses the classical mixture normal distribution.

Keywords: Probability distributions model · Iris segmentation · Mixture of probability distributions (MPD) · EM algorithm · CASIA iris images database · Markov property

1 Introduction

The iris of the human eye has been used as a biometric indicator to identify the human being similar to fingerprints. Iris pictures taken carefully in a lab are easily segmented and treated. However, in less constrained environments where iris images are captured at-a-distance and on-the-move, iris segmentation becomes much more difficult due to the effects of significant variation of eye position and size, eyebrows, eyelashes, glasses, contact lenses, and hair, together with illumination changes and varying focus conditions.

Iris recognition has been widely used as an indicator in the biometric system, [19]. Image segmentation is one of the important and difficult steps in image processing and analysis [2, 6, 7] appearing in many applications including pattern recognition, object detection, Medical imaging, etc. Recently, non-cooperative iris recognition has attracted much attention since it greatly extends the applicability of iris recognition [1, 3]. The image segmentation is an important early vision task where pixels with

A. Abraham et al. (eds.), *Proceedings of the 16th International Conference on Hybrid Intelligent Systems (HIS 2016)*, Advances in Intelligent Systems and Computing 552,
DOI 10.1007/978-3-319-52941-7_49

similar features are grouped into homogeneous regions (classes). The basis of any statistical pattern classification system is a parametric or nonparametric probability density of the feature vector of each class [5]. The segmentation task can be defined as the process of partitioning an image into some distinct classes. Each of such classes is homogenous with respect to the characteristic measured such as intensity, color, texture, [4]. Here, we are interested in statistical properties of the different classes, which are unknown and must be estimated. We can estimate characteristic parameters with iterative methods such as Expectation-Maximization (EM) algorithm [8]. Traditional iris processing by Daugman's integro-differential operator [9] extracts bi-nary features after mapping the textural area. In Hough transform, segmentation is an early technique that simply employs circular Hough transform [10] to find a parameterization of the boundaries needed for the mapping process.

In this paper, we propose a parametric method to estimate a mixture density using the EM algorithm. We introduce the mixture of exponential dispersion (MPD) model [13–16]. Note that the MPD is a generalization of Gaussian model. We can cite the following examples of MPD: Gaussian, Gamma, Pascal, Binomial, and Poisson. We propose to model different regions of iris image using mixture of probability distributions. The parameters of the mixture model are estimated using the Expectation-Maximization algorithm through maximum likelihood. Note that, we extend the EM algorithm by adding an update step to the M step, to improve the estimation of the parameters of our model. To confirm the efficiency and robustness of the proposed method, we apply our (MPD) modeling to the CASIA Iris images database [20]. We, then, compare our results to classical Gaussian approach.

The remainder of this paper is organized as follows: In the second section, we present the mixture of probability distributions model and the EM algorithm. In the third section, we apply our extended EM algorithm to segment the iris image into three regions using the Markovian and the independent pixel cases. We, Finally, discuss the results and conclude.

2 The Proposed Estimation and Methodology

2.1 Mixture of Probability Distributions

Finite Mixture Models provide powerful and flexible probabilistic models, which have been used successfully in many studies such as density estimation [17], clustering and segmentation.

The mixture density f can then be formulated as:

$$f(x) = \sum_{k=1}^{K} \pi_k f_k(x/\theta_k) \tag{1}$$

Where π_k is the mixing proportions such that: $0 < \pi_k < 1$ and $\sum_{k=1}^{K} \pi_k = 1$

The mixture model likelihood function is the probability of observing data for independent and Markovian sample (x_1, x_2, x_n). It is given by:

$$L = \prod_{i=1}^{N} \left(\sum_{k=1}^{K} \pi_k f_k(x/\theta_k)\right) \tag{2}$$

Where L (x_1, $x_2 \rightarrow x_N$, $\theta_1 \rightarrow \theta_k$ $\pi_{1,} \rightarrow \pi_k$).

The finite mixture parameters can be estimated very efficiently through maximum likelihood (ML) using the iterative expectation maximization (EM) algorithm [18]. Each iteration of the EM algorithm consists of two processes: the expectation or the E-step and the maximization or the M-step. The EM algorithm starts with initial searches of the mixture parameters. The E-step calculates the expected value of the Log (L) function using the current estimate for the parameters and the M-step computes parameters maximizing the expected log-likelihood found in the E-step.

2.2 Parameters Estimation Using EM Algorithm for Independent Data

Now, we will present our approach for estimating the parameters:

$$\theta = (\theta_1 \ldots \theta_k, \pi_1 \rightarrow \pi_k) \tag{3}$$

based on the extension of the EM algorithm. Let Y1…YN be a sample with mixture density f given by:

Step 1: Initialization

By K-Means, we calculate the initial parameters

$$\Theta^{(0)} = \left(\pi_1^{(0)}, \pi_2^{(0)}, \ldots, \pi_k^{(0)}, \theta_1^{(0)}, \theta_2^{(0)}, \ldots, \theta_k^{(0)}\right) \tag{4}$$

The expected value of the log-likelihood function, with respect to the conditional distribution given data under the current estimation of the parameters $\Theta^{(0)}$ is given by:

$$\mathbb{Q}\left(\Theta//\Theta^{(0)}\right) = \sum_{i=1}^{N}\sum_{j=1}^{K} T_{ij}^{(0)} \log(\pi_j + \sum_{i=1}^{N}\sum_{j=1}^{K} T_{ij}^{(0)} \log(fj(yi, \theta_j))) \tag{5}$$

Step 2: Expectation

With initial guesses for the parameters of our mixture model, we calculate the posterior probabilities ji for the observation y i in each component distribution fj By:

$$\begin{gathered} T_{ij}^{(0)} = \frac{\pi_j^{(1)} f_j(Y_i/\lambda_j^{(l)}, \theta_j^{(l)})}{\sum_{j=1}^{K} \pi_j^{(1)} f_j(Y_i/\lambda_j^{(l)}, \theta_j^{(l)})}; i = 1, \ldots, N; j = 1, \ldots, K. \\ \Theta^{(l+1)} = \arg\max_{\Theta} \mathbb{Q}\left(\Theta//\Theta^{(l)}\right) \end{gathered} \tag{6}$$

2.3 Parameters Estimation Using EM Algorithm for Markovian Data

Suppose that y i are dependent with density $f(yi, \theta_{ji})$

In addition, with link function

$$\theta_{ji} = \beta 0j\, y\, i\, -1 + \beta 1j + \beta 2j\, y\, i - 2 + \ldots + \beta rj\, y\, r - 2 = \psi\begin{pmatrix} u \\ ji \end{pmatrix} \tag{7}$$

$$\text{Where } \theta_{ji} = \psi\,(\mu_{ji}) \tag{8}$$

Suppose that each observation y i has r neighbors. The expected value of the log-likelihood function is given by:

$$\mathbb{Q}\left(\Theta//\Theta^{(0)}\right) = \sum\nolimits_{i=1}^{N} \sum\nolimits_{j=1}^{K} T_{ij}^{(0)} \log(\pi_j) + \sum\nolimits_{i=1}^{N} \sum\nolimits_{j=1}^{K} T_{ij}^{(0)} \left[\log fj\,(yi, \theta_{ji})\right] \tag{9}$$

By the same way, we calculate the initial parameters using K-mean algorithm, then in the expectation step, we calculate the posterior probabilities ij for the observation yi in each component distribution fj. Finally, in the (M) step, we Estimate the parameters.

$$\Theta^{(l+1)} = \arg\max_{\Theta} \mathbb{Q}\left(\Theta//\Theta^{(l)}\right) \tag{10}$$

To obtain a good segmentation process, our strategy is to go through four different steps. We remind that our work is based on supervised image segmentation. Therefore we assume that the eye image is composed of three different gray levels. First, we divide an eye image into three blocks of pixels by using K-Means method in order to obtain three regions. Second, we apply for each region, a Chi-square test to select the adequate density component (near to the empirical density component). In our case, each component density is selected from a list of probability distributions such as Gamma, Gaussian. The purpose of that choice is the fact that Gamma distribution is always positive and the Gaussian distribution is classically used as a reference method for comparative study. As a third step, we use the EM algorithm to estimate the parameters of our mixture model. Therefore, we propose a mixture of probability distributions that maximizes the likelihood function. We, then, segment and classify the image by using the estimated mixture model through the Bayes rule. The Process of the proposed segmentation generates a probabilistic mixture model by extending the EM Algorithm. The three regions (classes) of each eye image are denoted respectively by R1, R2, R3. Those regions are represented respectively by the function fk for k = 1, 2, 3.

Finally, we compare each original iris image with its empirical one. Our goal is to create a learning system in which our MPD algorithm is chosen for each different eye image. MPD can create more accurate mixture of probability distributions that nicely fit with each region. Therefore, each eye image can have different MPD for the different regions. In order to segment the iris, the component of our mixture algorithm varies from an eye image to another. First, the mixture algorithm should generate our CASIA data to assure the preliminary segmentation. The choice of the mixture components is dependent of the Chi-squared test. In our experimentation, we assume by default to use

Gamma and Gaussian distributions. In addition, in this work we consider both cases of environments: the independent and the Markovian environments.

3 Experimentation and Results

The problem of using the EM algorithm in image segmentation lies in the difficulty to estimate the number of components in the mixture. Many approaches, for example, the method used in [12] assumes that the number is known, this means that the user determines the number of segmentation regions in advance. In our case, the number of segmentation regions is fixed to three: iris, pupillary and spectra region or class (K = 3). The segmentation is carried out by assigning each pixel into a proper class according to the Bayesian rule. After the mixture identification, the Bayesian method is applied in order to classify the pixels according to their gray level x. Let $j(y)$ be the label of the class of the pixel y. defined by the Bayas formula.

$$j(x) = \arg\max_{1 \leq k \leq 3} \pi_k \gamma\left(x_i / a_k^n, b_k^n\right) \tag{11}$$

Image segmentation is an important phase of the image analysis [11]. A long-standing difficulty in the segmentation process lies in the fact that eye images are unique even in the same database. In other words, each eye image can contain different kind of noises, wish make it hard to efficiently segment all the images using just one model. Our objective in this paper is to carry out our MPD algorithm based on its adaptability to segment each different iris image and to get valuable results. In order to segment an iris image using the mixture of Normal and Gamma distributions model, we propose to divide the iris picture into three parts. This section illustrates the estimation procedure using the simulated data. First, we generate 100 observation iris images for testing. We, then, apply the two listed models for each image. Knowing that we divided the image into three parts, we estimate for each model the corresponding parameters. Each image is then segmented using the two models in both the markovian case and the independent case. This generates four different results for each test: Markovian Mixture of Normal, Markovian Mixture of Gamma- Normal distributions, independent Mixture of Normal and independent Mixture of Gamma-Normal distributions. We extend the General model from which we create the extended EM algorithm that we called earlier MPD. Our MPD is able to segment the iris by using the best mixture model for each different eye image.

The Chi-square test is used to obtain a learning system that facilitates the proposed statistical model of segmentation process. After that, we calculate the Kullback Distances (KL) between the distributions of the original image and the segmented image for each model in each case. The results are divided into four sets of Kullback distances (kullback-Leibler divergence) for every iteration. The formulas used to calculate the distance KL is given by:

$$\mathrm{KL} = (f||f^{\wedge}) = \int f(x) \log \frac{f(x)}{f(x^{\wedge})} dx \tag{12}$$

Where $\hat{f}$ is the estimated probability density and f is the true probability density. We, finally, compute the mean squared error (MSE) and we conclude.

In this section, we analyze and compare Markovian Mixture of Normal, Markovian Mixture of Gamma-Normal distributions, independent Mixture of Normal and Independent Mixture of Gamma-Normal distributions. In the independent environment, each pixel in the image is treated independently. In the other case, the Markovian environment, the density $f(yi,)$ of each pixel is dependant of its neighboring pixels. Giving an arbitrary weight (β) to each neighboring pixel, the density can then be presented as follows:

$$yi = \beta 1\, yi - 1 + \beta 2\, yi - 2 + \beta 3\, yi - 3 \tag{13}$$

To prove the performance of our MPD model verses the classical model, we propose to use as comparison tools mean squared errors MSE that is confirm MPD with Anova test and Kullback-Leibler distances. In the next part of work, we calculate the Kullback-Leibler distances between the empirical density and the estimated density of the image in the Markovian and independent environments. By observing the findings in (Fig. 1), first, we can mention that in both environments, the KL distance between the empirical and the estimated density by our mixture model is approximately lower than the estimates calculated by the Normal mixture model. Second, within the Markovian environment, the KL distance calculated by mixture of Gamma-Normal model distributions is less than the KL distance calculated by the mixture of Normal model distributions. In Fact, Our model gives excellent results due to the effects of the weighted neighboring pixels (Fig. 1). In a second step and in order to confirm the performance of the Mixture of Gamma-Normal distributions in the Markovian environment, we compare the mean squared error MSE given by our mixture model of each different set.

The Fig. 2 gives significant differences between the four models. The errors generated using MPD of Gamma-Normal model in both Markovian and independent environments are lower than errors by MPD Normal model in the same environment. Specifically, the lowest curve is Gamma exponential distribution algorithm in the Markovian environment. To confirm these results, we apply the Anova test to the Kullback-Leibler distances. The calculated Fisher test F is found lower than the critical value $F(2, 6274 < 3, 3534)$ with $P - \text{value} = 0, 019 < 0, 05$, which leads as to mention that the difference between the calculated F and the critical fisher F is significant and consequently there exists a clear variation between the Markovian and the independent models.

In the Markovian case, according to the MSE calculated for Mixture of Gamma-Normal distributions is slightly lower than MSE for Mixture of Gamma distributions $(0, 0029 < 0, 0030)$, which means that Mixture of Gamma distributions model is slightly better. The standard deviation results mean also that the segmentation systems using MPD algorithm tuned to specific CASIA image type may fail partially or completely whether in the markovian or the independent cases. This is explained when the classical algorithm behaves better than the MPD algorithm. The Fig. 3 gives examples of the few particular cases and confirm the accuracy of MPD algorithm. For this iris image dataset using the extended method mixture of distribution model, the segmentation

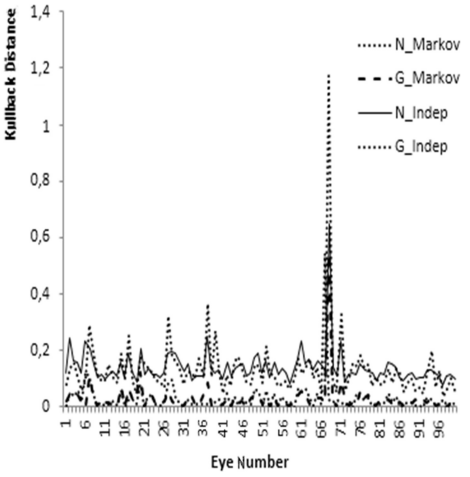


Fig. 1. Kullback-Leibler distances in both environments

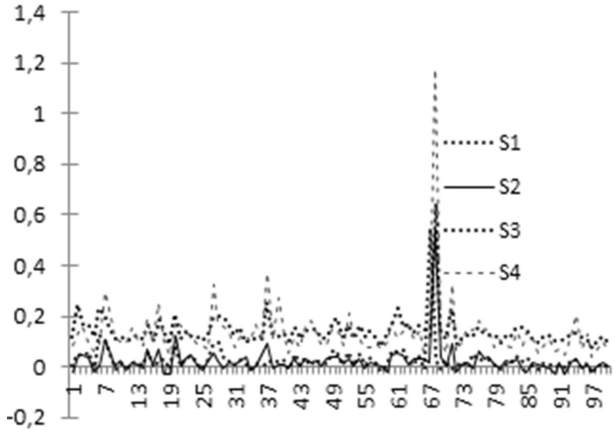


Fig. 2. MSE variation between classes

system is characterized by the creation of a specific mixture distribution for each eye image. In fact, due to the proposed learning system, each iris input will be associated to an adequate segmentation algorithm based on the specific mixture component. The MPD algorithm being tuned to deliver good results for this simulation, as this represents, the probabilistic segmentation method using the MPD algorithm is enhanced in the Markovian case. However, we cannot deny that the simulation process takes longer time. All different sets of probabilistic techniques of segmentation provide results by our new approach be confirmed. It is important to know that the classical method is also included in our new approach. In fact, the results of segmenting the iris image by using a mixture of pure Normal approach were valuable as well. Interestingly, MPD of gamma-Normal provides better results than MPD of Normal, even though it exhibits a systematic

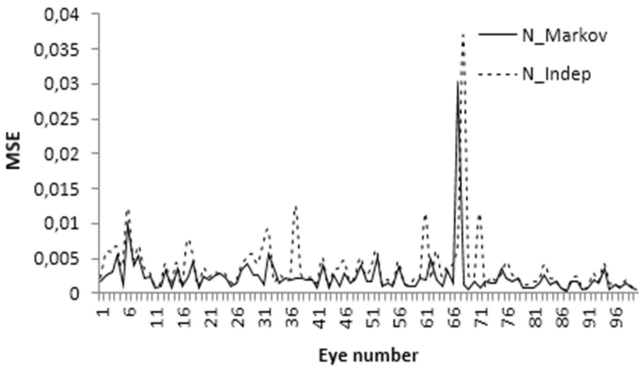


Fig. 3. Markovian and independent segmentation

exception. Typically, the robustness and the switch of the MPD algorithm to segment each iris differently from the same database can make a new concept of biometric segmentation. This concept is completely different from the classical segmentation concepts of Daugman's integro-differential and Hough transform.

4 Conclusion

The proposed method performs the segmentation of the divided regions of the iris from CASIA input images. The main idea is to create a robust and flexible algorithm to assure the best possible accuracy of the iris segmentation process. The performance of the proposed MPD algorithm is confirmed by the computation of the Kullback Leibler divergence. In addition, it is proved by Anova-Test. The calculation of the mean squared error is another proof to assure the performance of our study.

According to the results, we observe that the proposed MPD algorithm has better segmented the iris image than that obtained by the classical algorithm. The simulation of CASIA iris database using the proposed MPD algorithm has been applied for the Markovian segmentation case and the independent segmentation case. The results showed for both environments of work, the accuracy of the MPD algorithm using the mixture of Gamma-Normal distribution verses the mixture of normal distribution. As a perspective for future work, let us mention the possibility of testing the proposed algorithm by using other distributions such as Inverse Gamma, Pascal, Binomial, etc.

References

1. Bowyer, K.W., Hollingsworth, K., Flynn, P.J.: Image understanding for iris biometrics: a survey. Comput. Vis. Image Understand. **110**(2), 281–307 (2008)
2. Bergasa, L.M., Mazo, M., Gardel, A., Sotelo, M.A., Boquete, L.: Unsupervised and adaptive Gaussian s kin-color model. Image Vis. Comput. **18**, 987–1003 (2000)

3. Proenca, H., Alexandre, L.A.: Iris segmentation methodology for noncooperative Recognition. IEEE Proceedings of the Vision, Image and Signal (2006)
4. Zribi, M., Masmoudi, A., Hadrich, A.: Application of uniform B-Splines to unsupervised image segmentation, Signaux et Images de la Cote d'Opale (LISIC-EA 4491)
5. Hadrich, A., Zribi, M., Masmoudi, A.: A proposed nonparametric mixture density estimation using B-splines functions (2016)
6. Deng, H., Clausi, D.A.: Unsupervised image segmentation using a simple MRF model with a new implementation scheme. Pattern Recogn. **37**, 2323–2335 (2004)
7. Guo, G., Ma, S.: Bayesian learning, global competition and unsupervised image segmentation. Pattern Recogn. Lett. **21**, 107–116 (2000)
8. Dempester, A.P., Laird, N.M., Rubin, D.B.: Maximum likelihood from incomplete data via the E.M Algorithm. J. Roy. Stat. Soc. B **39**, 1–38 (1977)
9. Daugman, J.: How iris recognition works. IEEE Trans. Circiuts Syst. Video Technol. **14**(1), 21–30 (2004)
10. Wildes, R.P.: Iris recognition: an emerging biometric technology. In: Proceedings of the IEEE, vol. 85, pp. 1348–1363 (1997)
11. Uhl, A., Wild, P.: Multi-stage visible wavelength and near infrared iris segmentation framework. MultiMPDia Signal Processing and Security Lab-Department of Computer Sciences, University of Salzburg (2012)
12. Yamazaki, T.: Introduction of EM algorithm into color image segmentation. In: Proceedings of ICIRS 1998, pp. 368–371 (1998)
13. Barndorff-Nielsen, O.: Information and Exponential Families in Statistical Theory. Wiley, New York (1978)
14. Jorgensen, B.: Probability distributions models. J. Roy. Stat. Soc. **49**, 127–162 (1987)
15. Jorgensen, B.: The Theory of Dispersion Models. Chapman and Hall, New York (1997)
16. Letac, G.: Lectures on natural exponential families and their variance functions. Instituto de Matemática pura e Applicada, vol. 50. Monograf. Mat., Rio de Janeiro (1992)
17. Morris, C.N.: Natural exponential families with quadratic variance functions. Ann. Stat. **10**, 65–80 (1982)
18. Gauss, M., Marinho, C., de Andradeb, G.: Trans for MPD generalized linear models. J. Stat. Plan. Infer. **92**, 239–256 (2005)
19. Sruthi, T.K.: Literature review: iris segmentation approaches for iris recognition systems. Int. J. Comput. Eng. Res. **03**(5), 67–70 (2013)
20. Rajeev Kumar, M., Dilsath Fathima, M., Kiruthika, K., Saravanan, M.S.: Non-cooperative iris recognition: a novel approach for segmentation and fake identification, Department of IT, Vel Tech University, Avadi, Chennai, India (2013)

Chaff-Points Generation Using Knapsack Problem Resolution in Fingerprint Fuzzy Vault

Hachemi Nabil Dellys[1(✉)], Layth Sliman[2], Saliha Artabaz[1], Karima Benatchba[1], and Mouloud Koudil[1]

[1] Laboratoire de Méthodes de Conception de Systèmes (LMCS), Ecole nationale Supérieure en Informatique (ESI), BP 270 Oued-Smar, 160290 Algiers, Algeria
{h_dellys, s_artabaz, k_benatchba, m_koudil}@esi.dz
[2] Ecole d'Ingénieur Informatique et technologies du numérique (EFREI), 32 Rue de la république Villejuif, 94800 Paris, France
layth.sliman@efrei.fr

Abstract. In this paper, we introduce a new chaff-points generation method in fuzzy vault fingerprint. Our method is based on squares boundaries and Knapsack problem. Chaff-points generation process consists of creating a set of chaff points similar to authentic minutiae feature representation. Squares based chaff-points generation consist of delimiting a square boundary around each chaff-point in such a way that a square must never overlap with other squares. Unlike the typical squares method where chaff-points and their boundaries are generated sequentially, in our proposal, the referred chaff-points are generated all together then the knapsack problem formulation is used along with 0/1 dynamic programing resolution to determine if each chaff-point respects the constraints. The experiments results show that the chaff-points generation based on Knapsack method generates the referred number of chaff-points with reduced computing time, and especially, a smaller standard deviation in comparison to the sequential squares chaff-points generation.

Keywords: Fingerprint · Fuzzy Vault · Chaff-points · Minutiae representation · Composite representation · Knapsack problem · 0/1 Dynamic Programing

1 Introduction

Fuzzy vault is an error tolerate data securing method which uses public information. The method was proposed for the first time by Juels and Sudan [1] in 2002. The Fuzzy Vault method has been applied in biometrics to secure templates. The conventional encryption techniques used to secure templates does not work for securing biometric templates because different capture of the same modality gives different data. As a result, the use of other methods, like Fuzzy Vault, becomes necessary. In this paper we focus in fingerprint Fuzzy Vault [2–5, 29].

Fuzzy vault is a process composed on two phases which is encoding phase and decoding phase. In decoding phase, the fingerprint features and secret key are encrypted using a secret key, this phase is composed of five stages:

A. Abraham et al. (eds.), *Proceedings of the 16th International Conference on Hybrid Intelligent Systems (HIS 2016)*, Advances in Intelligent Systems and Computing 552,
DOI 10.1007/978-3-319-52941-7_50

1. Compute the Characteristic Polynomial [6, 7, 9].
2. Fingerprint Feature Representation [10–12].
3. Points Modelling [7, 12, 15].
4. Chaff Points Generation [19, 20, 22, 24, 30].
5. And Vault Construction and Storage [16].

In the decoding phase, we try to reconstruct the secret key using fingerprint features and the encoded vault, this phase is composed of three stages:

1. Points alignment [13–16, 23, 26].
2. Determination of Correspondence Set [11, 17, 18].
3. Secret Polynomial Reconstruction [10, 19].

The security level depends on feature representation and strategies of chaff-points generation. Thus, several chaff-points generation techniques are proposed like one and two thresholds methods [26, 27], composite representation [5, 31] and squares method [12]. Experiments conducted in [16] shows that composite representation is more efficient than the other representations because it does not require a point alignment stage in decoding phase. In the other side, the paper shows that squares method generates the same number of the referred chaff-points with less computing time in comparison to other methods. In our work, we choose to use composite representation and squares method.

Classical squares method is based on generation of similar size squares boundary around the points. A new point is added only if its boundary is not overlapping with boundaries of others points in the vault. For this, chaff-point is generated one per one using random thresholds distance.

In this paper, we propose to use the Knapsack problem to generate chaff-points. All referred chaff-points are generated randomly using composite representation and squares method. Afterwards, we use a 0/1 dynamic programming algorithm to determine if each generated point respects the aforementioned overlapping constraint.

In the following, we present different chaff-points generation methods in fingerprint fuzzy vault in Sect. 2. Section 3 details our methods of chaff points generation using knapsack problem. In Sect. 4 we describe the comparison experiments. At the end a conclusion and future work is explained.

2 Chaff Points Generation

Chaff-points in fingerprint is a misleading point similar to the authentic minutiae representation. These points are generated randomly in order to hide authentic minutiae. Chaff-points generation is subject to two constraints [23]:

i. A chaff-point must not be very close to an authentic point;
ii. Two chaff-points should not be very close. Otherwise, the Chaff points will be easily detected by the attacker.

In the other side, the security level increase when number of chaff-points is greater, but the size of memory space used by of the vault [24] and computing time [26]

becomes very great. Furthermore, the freedom degree of the chaff-points increases significantly, and an attacker can easily detect authentic points [25].

Thus, a compromise between security, storage space and computing time is to determine the number of the generated chaff point according to the security level that we need, taking into consideration that a good security level of biometric system requires a chaff-points number greater than authentic points number [28].

In fuzzy vault process, each stage depends closely on those above. For it, we choose to use composite representation [16] in this paper to have the best performance, because this representation does not require alignment point stage. Below we present briefly different fingerprint minutiae representation then we present most important chaff-points generation methods.

2.1 Fingerprint Feature Representation

There are several features that can be extracted from fingerprint, only two of them can be used in biometric recognition: minutiae representation and descriptors representations [18, 33]. Minutiae representation is the most used one. This representation uses the minutiae of fingerprint to generate a template. We present the main fingerprint minutiae representation in following:

Minutiae representation by tuple uses a set of coordinates composed only by Cartesian coordinates (x, y) [6]. Another representation adds orientation θ (x, y, θ) [9, 20]. Finally, the most used is representation in which the type of minutiae (endpoint or fork) is added to the tuple (x, y, θ, T) [10, 21]. This type of representation is easy to implement but requires alignment points stage which generates several alignment error.

Another type of minutiae representation is representation of a minutia by using its neighbours. The main used techniques in this representation are the five nearest neighbours structure [11], Voronoï neighbours [11] and composite representation [17, 34] which we use in this paper. In the composite representation, the minutia (*mi*) is described by a 3-tuple ($\boldsymbol{d_{i-j}}$, $\boldsymbol{\varphi_{i-j}}$, $\boldsymbol{\theta_{i-j}}$) using a minutia (*mj*) in their neighborhood where:

$\boldsymbol{d_{i-j}}$: is the Euclidian distance between *mi* and *mj*;
$\boldsymbol{\varphi_{i-j}}$: is the difference between orientation angle of *mi* and *mj*;
$\boldsymbol{\theta_{i-j}}$: is the counter-clockwise angle between the orientation of *mi* and direction from *mi* to *mj*.

2.2 Chaff Points Generation Methods

Chaff-points are represented by two tuples structure named historically abscissa (*c*) and ordinate (*d*). Abscissa is a data structure similar that minutiae representation, and ordinate is an evaluation of chaff-points using characteristic polynomial to avoid a bad interpolation in decoding phase. Abscissae and ordinates are generated by different methods; we present the main as following:

A. Chaff points abscissae generation
The main chaff-points abscissae generation are:

i. One threshold method: This strategy generates points using Euclidean distance to separate each points at least by threshold (δ) (Fig. 1-a) [26, 27].
ii. Two threshold method: this strategy takes into account the distribution of the authentic points that can be fairly close together and formed a points mass. An attacker can distinguish this mass to find authentic points (Fig. 2-b). To resolve this problem a second threshold are used. The first distance ($\delta 1$), separate chaff-point from others, the second distance ($\delta 2$), separate chaff-points to authentic points [26, 27].
iii. Squares based method: This method create boundary around each point of the vault by a fixed size square, so squares must never overlap [20, 31].
iv. Cells based method: This is a special case of squares method [29]. The plan is divides into fixed-size cells where each authentic point must be into a separate cell, after chaff-points are generates randomly into each empty cell. But this method is easily circumvented because the cells are pre-set in advance.

B. Chaff points ordinates generation
Chaff-points ordinates are generating randomly after checking the constraint saying that (c_i, d_i) is not on the secret polynomial [6, 10, 18, 19, 21], or by taking $d_i = P(u_i) + \alpha$, where P is a secret polynomial, α is a real generated randomly and u_i is locking unit obtained from c_i [30].

3 Related Work

Chaff-points generation is one of more important stages in fingerprint fuzzy vault process. So, many contributions are proposed to generates chaff-points efficiency while meeting the constraints mentioned earlier. First propositions [26, 27] generates chaff-points randomly with respect a certain threshold. One threshold method considered only one threshold between each points (authentic and chaff), but authentic points can be fairly close together and formed a points mass, an attacker can distinguish this mass to find authentic points. Two thresholds method solve relatively this problem by introducing a second threshold. The first threshold is used between chaff-points only and the second threshold is used between chaff-points and authentic points.

But chaff-points generation is closely related with which feature representation is used. According to [16], composite representation likes the best because this representation not need to make points-alignment stage which is the stage that take greatest computing time. The same work [16] shows that thresholds method generates a limited chaff-points (around 200 chaff-points for one threshold method and 350 chaff-points for two thresholds method), And computing time increase exponentially. Thus, other methods are proposed like squares boundaries based generation [12, 20, 31] and cells based generation [29]. In [16], a comparative study of these methods shows that squares boundaries based generation method generates exactly the referred number of chaff-points and computing time is significantly reduced.

Classical squares based method [12] generates chaff-points sequentially, this method is interesting but computing time can be ameliorating if an alternative approach than sequential is used. Thus, we propose a new chaff-points generation approach.

4 Chaff -Points Generation Using Knapsack Problem

In this section we present our solution of Chaff-points generation using Knapsack problem formulation and 0/1 dynamic programming algorithm.

4.1 Knapsack Problem

Knapsack problem is one of the most known problem in combinatorial optimization. The original problem is stated as follows:

A traveller has a knapsack with maximum containing weight W, and there is some object with weight w_i and value v_i that traveller wants to carry in the knapsack. The traveller should carry the maximum number of valuable items without exceeding the maximum containing weight W.

Mathematic formulation of this problem is:

$$\begin{aligned} &\forall i, x_i \in \{0,1\} \text{ Maximize}(\textstyle\sum_n^{i=1} v_i x_i) \\ &\text{Subject to: } \sum_n^{i=1} w_i x_i \leq W \end{aligned} \tag{1}$$

Where $x_i = 1$, if item i is put in knapsack, otherwise $x_i = 0$.

Knapsack problem has several variant, one of them is when $w_i = v_i$ then we obtain a special case of knapsack problem named subset sum problem, which may be formulated as follows:

$$\begin{aligned} &\forall i, x_i \in \{0,1\} \text{ Maximize}(\textstyle\sum_n^{i=1} v_i x_i) \\ &\text{Subject to: } \sum_n^{i=1} w_i x_i \leq W \end{aligned} \tag{2}$$

Knapsack problem can be resolved using 0/1 Dynamic programming (DP). We can formulate recursive resolution as follows:

Let KS(i, W) be a value of optimal solution, then:

$$KS(i, W) = \begin{cases} 0 & \text{if } i = 0 \text{ or } W = 0 \\ KS(i-1, W) & \text{if } wi > W \\ Max(KS(i-1, W), & KS(i-1, W - wi) + wi) \end{cases} \tag{3}$$

4.2 Chaff-Points Generation Formulation Using Knapsack Problem

Knowing that we use composite representation ($\boldsymbol{d_{i-j}}$, $\boldsymbol{\varphi_{i-j}}$, $\boldsymbol{\theta_{i-j}}$) and squares method, we can adapt the mathematic formulation of subset sum problem to chaff-point generation as follows:

$$\begin{aligned} &\text{Maximize}\left(\textstyle\sum_{n}^{i=1} P_{\mathrm{i}}\mathrm{c_i}\right) \\ &\text{Subject to:} \quad \sum_{n}^{i=1} P_{\mathrm{i}}\mathrm{c_i} \leq \mathrm{nbChaff} \\ &\begin{cases} \forall i, (0,0) < P_i(x,y) < (X_{max}, Y_{max}) \\ \forall i, B_{i-1}(X_{min}) < P_i(x) < B_{i-1}(X_{max}) \\ \forall i, B_{i-1}(Y_{min}) < P_i(y) < B_{i-1}(Y_{max}) \end{cases} \end{aligned} \tag{4}$$

Where:

- P_i, ith point generated
- $\forall i,\ c_i \in \{0,1\}$
- X_{max} Maximum value of Euclidian distance (first tuple)
- Y_{max} Maximum value of the difference between orientation (second tuple)
- $B_i(X_{min})$ Minimum value of euclidian distance for Boundary of ith point generated
- $B_i(X_{max})$ Maximum value of euclidian distance for Boundary of ith point generated
- $B_i(Y_{min})$ Minimum value of difference between orientation for Boundary of ith point generated
- $B_i(Y_{max})$ Maximum value of difference between orientation for Boundary of ith point generated

4.3 Chaff-Point Generation Resolving Using 0/1 DP Algorithm

By using 0/1 DP algorithm to chaff-points generation with knapsack formulation we obtain the following algorithm:

- **Step 1:** we initialize the vault with authentic points and generates their squares boundaries.
- **Step 2:** For each point (P_i) in the vault, generates randomly four chaff-points and their boundaries in neighbourhood, and put $c_i = 0$;
- **Step 3:** Run 0/1 DP algorithm to verify if each point verifies the constraints. If $c_i = 0$ and the constraints of P_i are verified and then $c_i = 1$;
- **Step 4:** If number of chaff-points referred are generated or no more points can be generating than end. Else, return to step 2.

5 Experimental Results

In this section, we conduct an experimental comparison between squares generation with knapsack problem resolution and classical algorithm of squares method as proposed in [12].

First, we discard thresholds methods because, as shown in Fig. 1, their computing time increase significantly after 150 generated chaff-points and the number of chaff-points does not exceed 300 chaff-points [16]. However, the squares method achieves referred chaff-points number. Therefore, we focus our comparison on chaff-points generation using classical squares method and our adapted generation techniques based on knapsack problem. In this paper, we use the notation of squares method to indicate classical squares method [12], and Knapsack method to indicate our proposal.

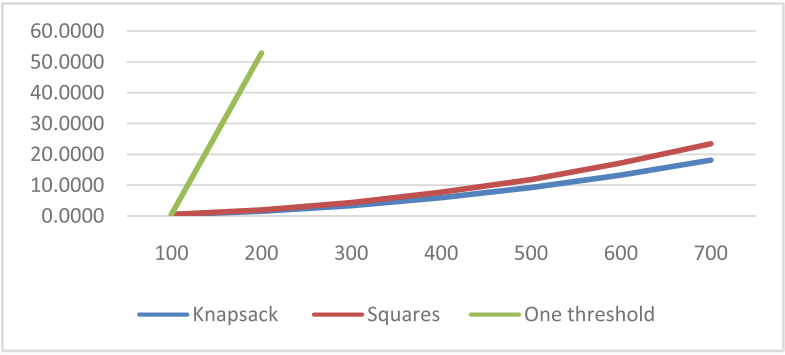


Fig. 1. Computing time of chaff-points generation methods.

The experiment is launched in the DB2_A, FVC2006 database with 140 fingers and 12 samples per finger. The experiment was run using a software platform based on Java and Matlab, and a computer hardware based on an Intel Quad-Core processor and 16 GB of Random memory.

Table 1. Average, standard deviation, minimum and maximum time computing par chaff-points generated

Chaff-points number	Knapsack method				Squares method			
	Avg[a]	Std[b]	Min	Max	Avg	Std	Min	Max
100	0,39	0,01	0,37	0,45	0,48	0,14	0,35	1,18
200	1,50	0,02	1,45	1,68	1,94	0,57	1,41	4,55
300	3,36	0,04	3,28	3,63	4,36	1,29	3,11	10,13
400	5,96	0,06	5,77	6,46	7,70	2,21	5,59	18,33
500	9,21	0,08	8,96	10,98	11,82	3,43	8,78	25,60
600	13,26	0,10	13,00	15,66	17,17	5,08	12,41	44,39
700	18,12	0,16	17,56	19,81	23,42	6,77	16,94	57,46

[a] Average
[b] Standard deviation

Since squares method and Knapsack method generate the same number of chaff-points, we focus our comparison in the computing time. Table 1 shows the average of computing time of each method to generate a number of chaff-points included between 100 and 700 points. Clearly, the computing time using knapsack method is less than this of squares method. Furthermore, the more chaff-points are generated, the greater the difference in computing time becomes.

We can note that the standard deviation (Fig. 2), in knapsack method is low and increases slightly (i.e. it is 0,01 for 100 chaff-points generated to reach only 0,16 for 700 chaff-points generated).

Generally, the computing time of squares method is often less than that of knapsack method if the number of generated points is small. However, unlike squares method computing time which increases exponentially in relation to the number of generated points, knapsack method keeps a rather stable computing time whatever the number of the generated points is.

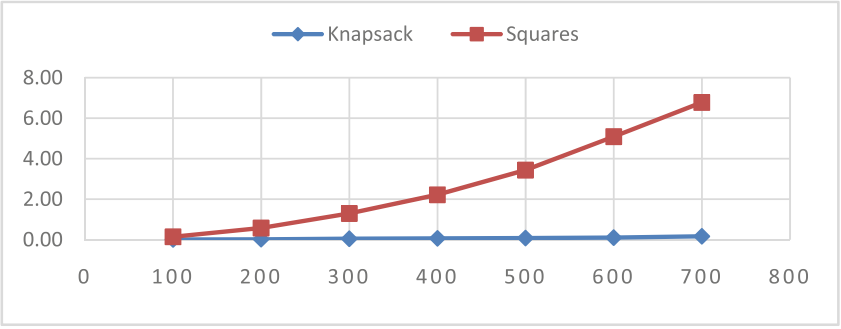


Fig. 2. Standard deviation evolution

6 Conclusions

In this paper, we proposed a new chaff-point generation method based on squares boundaries and knapsack problem. Unlike to classical squares method of chaff-points generation when chaff-points and their squares boundaries are generated sequentially with checking the constraints of non-overlapping between squares in each generation, we generate all referred points randomly. then we use knapsack problem formulation and 0/1 dynamic programing resolution to determine if each point satisfies the constraints or not. The process is repeated until the number of chaff-points is reached. We focus our comparison only in squares boundaries based method because the thresholds methods generate a limited number of chaff-points and take a great computing time. In the other hand, squares based method generates exactly the referred number of chaff-points. We showed also that computing time is lower in knapsack based solution. In addition, this computing time average grows when the number of chaff-points grows.

Another important finding is that the value of standard deviation of Knapsack based method is still small, which means that the computing time is relatively the same for each sample of finger. Unlike in classical squared method for which the standard deviation is relatively great and hence this number grows significantly when the number of chaff-points grows.

Finally, this work is a part of a fuzzy vault process. Other works should be conducted in different stages to obtain the most efficient rate of false rejection rate (FRR) and false acceptance rate (FAR) with the lowest computing time.

References

1. Sudan, J.A.M.: A fuzzy vault scheme. In: Proceedings of the 2002 IEEE International Symposium on Information Theory (2002)
2. Nguyen, T.H., Wang, Y., Ha, Y., Li, R.: Performance and security-enhanced fuzzy vault scheme based on ridge features for distorted fingerprints. IET Biometrics **4**(1), 29–39 (2015)
3. Limshe Pearce, A., Mala, R., Nirmala, A., Chandru, B.: Security enhancement for finger print using double encryption and fuzzy vault scheme. Int. J. Adv. Res. Comput. Commun. Eng. **5**(3), March 2016
4. Joshi Ram, B.: Cryptographic fuzzy vault with image processing. Int. J. Adv. Res. Comput. Commun. Eng. **5**(1), January 2016
5. Dellys, H.N., Benadjimi, N., Boubakeur, M.R., Sliman, L., Benatchba, K., Artabaz, S., Koudil, M.: A critical comparison of fingerprint fuzzy vault techniques. In: Badioze Zaman, H., Robinson, P., Smeaton, A.F., Shih, T.K., Velastin, S., Jaafar, A., Mohamad Ali, N. (eds.) IVIC 2015. LNCS, vol. 9429, pp. 178–188. Springer, Heidelberg (2015). doi:10.1007/978-3-319-25939-0_16
6. Uludag, U., Jain, A.: Securing fingerprint template: fuzzy vault with helper data. In: 2006 Conference on Computer Vision and Pattern Recognition Workshop, CVPRW 2006. IEEE (2006)
7. Juels, A., Sudan, M.: A fuzzy vault scheme. Des. Codes Crypt. **38**(2), 237–257 (2006)
8. Zhou, R., et al.: Adaptive sift-based algorithm for specific fingerprint verification. In: 2011 International Conference on Hand-Based Biometrics (ICHB). IEEE (2011)
9. Jeffers, J., Arakala, A.: Fingerprint alignment for a minutiae-based fuzzy vault. In: 2007 Biometrics Symposium. IEEE (2007)
10. Lee, S., et al.: Analysis of tradeoffs among verification accuracy, memory consumption, and execution time in the GH-based fuzzy fingerprint vault. In: 2008 International Conference on Security Technology, SECTECH 2008. IEEE (2008)
11. Choi, W., et al.: Apparatus and method for polynomial reconstruction in fuzzy vault system, Google Patents (2012)
12. Khalil-Hani, M., Marsono, M.N., Bakhteri, R.: Biometric encryption based on a fuzzy vault scheme with a fast chaff generation algorithm. Future Gener. Comput. Syst. **29**(3), 800–810 (2013)
13. Nguyen, T.H., et al.: A fingerprint fuzzy vault scheme using a fast chaff point generation algorithm. In: 2013 IEEE International Conference on Signal Processing, Communication and Computing (ICSPCC). IEEE (2013)
14. Nandakumar, K., Nagar, A., Jain, A.K.: Hardening fingerprint fuzzy vault using password. In: Lee, S.-W., Li, S.Z. (eds.) ICB 2007. LNCS, vol. 4642, pp. 927–937. Springer, Heidelberg (2007). doi:10.1007/978-3-540-74549-5_97

15. Moon, D., et al.: Implementation of automatic fuzzy fingerprint vault. In: 2008 International Conference on Machine Learning and Cybernetics. IEEE (2008)
16. Dellys, H.N., Benadjimi, N., Boubakeur, M.R., Sliman, L., Ali, F.: Chaff point generation by squares method using composite representation in fingerprint fuzzy vault. J. Inf. Assur. Secur. **11**(1), 1–10 (2016). 10 p.
17. Xi, K., Hu, J.: Biometric mobile template protection: a composite feature based fingerprint fuzzy vault. In: 2009 IEEE International Conference on Communications, ICC 2009. IEEE (2009)
18. AlTarawneh, M., Woo, W., Dlay, S.: Fuzzy vault crypto biometric key based on fingerprint vector features. In: 2008 6th International Symposium on Communication Systems, Networks and Digital Signal Processing, CNSDSP 2008. IEEE (2008)
19. Dang, T.K., Nguyen, M.T., Truong, Q.H.: Chaff point generation mechanism for improving fuzzy vault security. IET Biometrics **5**(2), 147–153 (2016)
20. Park, U., Pankanti, S., Jain, A.: Fingerprint verification using SIFT features. In: SPIE Defense and Security Symposium. International Society for Optics and Photonics (2008)
21. Khachatryan, G., Aram, J., Khasikyan, H.: Alignment-free fuzzy vault scheme for fingerprints (2013)
22. Rahul, H., Manavjeet, K.: Novel chaff generation for fingerprint fuzzy vault. Br. J. Math. Comput. Sci. **10**(3), 1–9 (2015)
23. Örencik, C., et al.; Improved fuzzy vault scheme for fingerprint verification (2008)
24. Nagar, A., Nandakumar, K., Jain, A.K.: Securing fingerprint template: fuzzy vault with minutiae descriptors. In: 2008 19th International Conference on Pattern Recognition, ICPR 2008. IEEE (2008)
25. Krivokuća, V., Abdulla, W., Swain, A.: A dissection of fingerprint fuzzy vault schemes. In: Proceedings of the 27th Conference on Image and Vision Computing New Zealand. ACM (2012)
26. Sood, P., Kaur, M.: Methods of automatic alignment of fingerprint in fuzzy vault: a review. In: 2014 Recent Advances in Engineering and Computational Sciences (RAECS). IEEE (2014)
27. Nandakumar, K., Jain, A.K., Pankanti, S.: Fingerprint-based fuzzy vault: implementation and performance. IEEE Trans. Inf. Forensics Secur. **2**(4), 744–757 (2007)
28. Alibeigi, E., Rizi, M.T., Behnamfar, P.: Pipelined minutiae extraction from fingerprint images. In: 2009 Canadian Conference on Electrical and Computer Engineering, CCECE 2009. IEEE (2009)
29. Uludag, U., Pankanti, S., Jain, A.K.: Fuzzy vault for fingerprints. In: Kanade, T., Jain, A., Ratha, N.K. (eds.) AVBPA 2005. LNCS, vol. 3546, pp. 310–319. Springer, Heidelberg (2005). doi:10.1007/11527923_32
30. Clancy, T.C., Kiyavash, N., Lin, D.J.: Secure smartcard based fingerprint authentication. In: Proceedings of the 2003 ACM SIGMM Workshop on Biometrics Methods and Applications. ACM (2003)
31. Yang, S., Verbauwhede, I.: Automatic secure fingerprint verification system based on fuzzy vault scheme. In: 2005 Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, (ICASSP 2005) (2005)
32. Jeffers, J., Arakala, A.: Minutiae-based structures for a fuzzy vault. In: 2006 Biometrics Symposium: Special Session on Research at the Biometric Consortium Conference. IEEE (2006)
33. Harmer, K., et al.: Fuzzy vault fingerprint smartcard implementation using an orientation-based feature vector. In: BLISS (2008)

Adaptive WSNs Based on HW/SW Implementation of Selective Relaying Communication Technique

Nesrine Atitallah[1,2(✉)], Kais Loukil[1,2], Hela Hakim[2,3], Abdelfattah Obeid[4], and Mohamed Abid[1,2]

[1] CES Laboratory, National Engineering School of Sfax, Sfax, Tunisia
{nesrine.atitallah,kais.loukil,mohamed.abid}@ceslab.org
[2] Digital Research Center of Sfax (CRNS), Sfax, Tunisia
[3] COSIM Laboratory, Higher School of Communication of Tunis, Tunis, Tunisia
hela.hakim@supcom.tn
[4] National Electronics, Communication and Photonics Center, KACST, Riyadh, Saudi Arabia
obeid@kacst.edu.sa

Abstract. Power consumption poses a challenge issue in Wireless Sensor Networks as the sensor node is power-constrained. Since Wireless communication is among the most energy expensive tasks, many power efficient transmission strategies are reported in the literature. However, most of them are purely software-based implemented. Since relaying algorithms are intensive-computing tasks, a purely software-based implementation is not the wise choice. This work comes to propose an alternative choice for implementing our earlier proposed relaying algorithm, which is a HW/SW implementation. To do so, we have identified the greedy CPU time and energy consuming tasks and implemented their dedicated hardware accelerators. We prototype our work using an FPGA board. Design details and interfacing of the different accelerators show important time execution and power consumption reduction of our algorithm.

Keywords: Hardware accelerator · Interfacing · Profiling · Power consumption · Selective relaying · Wireless Sensor Networks

1 Introduction

Because of the increasing availability and maturity of the underlying technology, Wireless Sensor Networks (WSNs) are becoming increasingly popular in the use of many applications including scientific research, healthcare, environmental monitoring, security and surveillance, among others [1]. They consist of scattered sensor nodes essentially in charge of monitoring the surroundings area and communicating with each other. Since the sensor nodes have limited resources, WSNs are inherently constrained. This is the reason why energy consumption without

A. Abraham et al. (eds.), *Proceedings of the 16th International Conference on Hybrid Intelligent Systems (HIS 2016)*, Advances in Intelligent Systems and Computing 552,
DOI 10.1007/978-3-319-52941-7_51

decreasing the networks functionalities is one of the major issues in WSNs. Indeed, minimization of power consumed by each sensor node in WSN is the crux of the matter. As a result, WSNs adaptation and reconfiguration represent ideal solutions to efficiently manage limited resources regarding conditions and context changes [2]. In other words, there are different threatening problems that need an adaptive WSN such as regular code updates, flexibility in adopting energy and performance, efficient wireless communication link or interface, etc. For these reasons, sensor node must adapt its mode of functioning either periodically or using a trigger event to ensure the best use of available resources.

This paper investigates adaptive WSNs that allow a sensor node to change its energy consumption in ways that increase its lifetime which has a direct impact of the entire network lifetime, as well. Indeed, many adaptive WSN platforms have been proposed to support different applications with main focus on low power consumption [3–7]. A typical sensor node is built around a main processor, a local bus, a main memory, and standard components (like hardware counters and Input/Output I/O peripherals). However, when executing intensive tasks such as routing protocols, cryptography algorithms and data communications on moderate known motes, the execution time is considerably high and will deplete certainly the node battery. In order to satisfy the energy constraints, the hardware node design can be enhanced with specialized hardware computing units such as hardware accelerators, coprocessors and reconfigurable units.

In this respect, Field Programmable Gate Array (FPGA) represents a very suitable candidate for high performance WSN applications thanks to its flexibility and high speed of data processing. Moreover, this reconfigurable device allows an implementation with custom logic which is significantly faster than software. Therefore, when executing a task on FPGA, energy efficiency can be achieved better than it is on Microcontrollers (MCUs) or core processors. That is the reason why several authors [3–7] used different FPGA families in the node architecture. In fact, the use of FPGA in research nodes is either to accelerate different routing and security algorithms or to partially reconfigure the node architecture. In [4], the FPGA is adopted to accelerate data processing and JPEG image compression algorithm with less power consumption in the Wireless Multimedia Sensor Network (WMSN) sensor node. Furthermore, in [6], authors presented a reconfigurable node based on a low-cost FPGA (Spartan-3, XC3S200). They also proposed a support middleware for the real-time configuration. It is basically a module that resides in the node to reconfigure it according to the application needs. We propose using the FPGA, as reconfigurable component, which performs complex architecture, operates in low-power and has the ability to total or partial dynamic reconfiguration of the architecture.

Throughout this paper, our concern will focus on hardware adaptation by performing the most complex application by dedicated hardware instead of the microprocessor. In [10], authors presented a reconfigurable analog/mixed-signal sensing platform to improve energy efficiency. They added signal-processing blocks and circuit components at the hardware level to reduce the activity of the MCU and the radio. Furthermore, to reduce the power consumption, authors

in [8] proposed a hardware-based and reconfigurable network accelerator which is prototyping on Complex Programmable Logic Device (CPLD) board. In this work, we address the HW/SW implementation of a new relay selection algorithm proposed in an earlier work [11]. According to our related work study, relaying algorithms in WSN are purely software-based implemented. Software-based implementation is simpler than the hardware one, however it consumes much more energy and execution time. Since, relaying algorithms are intensive-computing tasks, a purely software-based implementation is not the wise choice. This work comes to propose an alternative choice for implementing our earlier proposed relaying algorithm by proposing a new implementation method based on both software and hardware. To do that, we have decomposed our relaying algorithm on elementary tasks. Then, we have identified the greedy CPU time and energy consuming tasks. After that, these greedy consuming tasks are implemented as HW accelerators, which we have designed. In this way, we have reduced the overall execution time and energy consumption of our algorithm. We prototyped our earlier proposed relaying algorithm using an FPGA board. Therefore, HW/SW implementation provides higher levels of performance with reduced energy consumption than the pure software equivalent. We also interface different accelerators and test the architecture of a platform that uses reconfigurable hardware to accelerate the selective relaying algorithm.

The remainder of this paper is organized as follows. Section 2 provides an overview about the investigated relay selection algorithm. Section 3 presents the HW/SW implementation of the algorithm. Moreover, the performance of the proposed strategy is evaluated and power consumption at run-time is measured. Finally, Sect. 4 draws out concluding remarks.

2 Relay Selection Algorithm

In this section, a brief description of the Relay selection algorithm is given. Then, a detailed analysis is provided to better choose the appropriate hardware implementation modules.

2.1 Relaying Protocol Description

The relay selection algorithm aims to select the power efficient path between any source-destination pair in the cluster [11]. The goal of this approach is to minimize the transmit power energy based on Bit Error Probability (BEP) criterion for Selective Digital Relaying (SDR) [9]. We assume a number N of nodes, randomly distributed within a cluster, whose the coordinates x and y are between $[-a, a]$ and $[-b, b]$, respectively. So, the position and the distance between each others are known. We suppose that a source node S transmits information to a destination node D. The source S and the destination D communicate over a channel with a slow and frequency-flat Rayleigh fading coefficient. So, the transmitted signal propagates through different channels before arriving to its destination. Therefore, a relay R can collaboratively forward the signal, received from

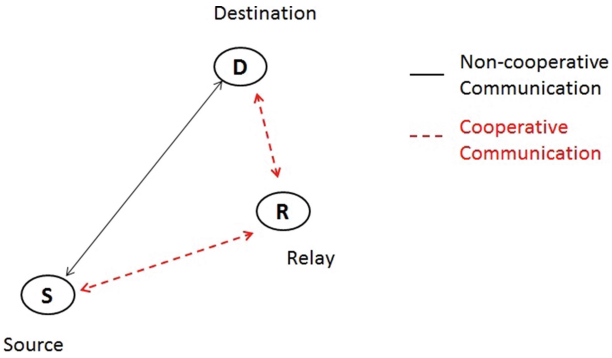


Fig. 1. Different communication systems

tha source S, to D in the second stage using either Amplify-and-Forward (AF) or Decode-and-Forward (DF) relaying strategy, as illustrated in Fig. 1. After that, received signals by the destination D are combined using the Maximum Ratio Combining (MRC) technique. In other words, a source node S transmits a signal to destination while the $N-2$ remaining nodes listen. Then, the activated relay retransmits the source signal to the destination with the least transmit power computed by the relay selection algorithm. The BEP expressions, derived in [11], are used to compute the minimum transmit power in cooperative link as well as in direct link. Thus, it can efficiently allocate the least transmission power among the source and the relay that satisfy a predefined level of reliability while decreasing the energy consumption of the node. Before sending symbols to the destination D, the latter executes the relay selection algorithm whose main steps are:

1. Generation of the distance and SNR matrices. This step is done within Generation of Distance Matrix and Generation of SNR Matrix functions.
2. Selection of the set of reliable nodes having a received source-node Signal to Noise Ratio (SNR) above a predefined threshold value, noted γ_{th}. This step is done within the Reliable Nodes Set identification function.
3. Calculation of the minimum transmit power to be used in the cooperative link as well as in the direct link. This step is done within the Minimum transmit Power in Cooperative Link function and Minimum transmit Power in Direct Link function.
4. Selection of the optimal relay will offer the least transmit power combination in source-destination and relay-destination links while maintaining the required level of reliability $P_{e_{th}}$.
5. Comparison between bit power transmission to be used in the direct link as well as in the cooperative link to decide which communication scheme to be used in the transmission process.
6. Communication of the source and the relay (in case of cooperative communication) with the appropriate transmit power to be used in the transmission process, as depicted in Fig. 2.

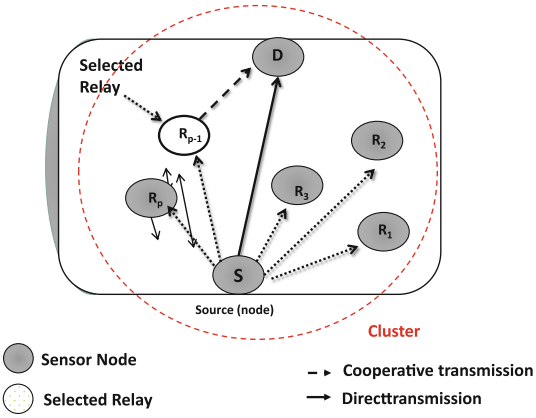


Fig. 2. Relay selection scheme

The flowchart in Fig. 3 summarizes the main steps achieved by the destination to select the efficient-path to be chosen by the source S to send information, as explained in the previous paragraph.

2.2 Relay Selection Algorithm Analysis

The purpose of this subsection is to identify the candidate functions for hardware implementation. More has an implementation accelerator, it will be more efficient (less execution time) at the expense of a more complex architecture and thus consumes more energy. As there is a tradeoff between energy consumption and performance, it is important to select the efficient solution. Through profiling and complexity analysis of each function of the entire algorithm, critical functions that require hardware implementation are identified.

In fact, profiling technique and high level complexity analysis give the designer statistics on the different functions performed in the software implementation of the relay selection algorithm (execution time, percentage of processor occupation, number of iterations, memory usage, etc.). Therefore, functions that take more execution time will be ideal candidates for a hardware implementation. In our work, we used a direct measurement on the map for the accuracy of this method. Indeed, once the different hardware configurations are shipped, it can be used to make the measurements and this through a software "timer". It acts as a down counter which decrements every p clock ticks which is called period. It is fixed in the design of the hardware part. To measure the execution time of a task, we read the value of the timer before and after its execution. The time taken for this period is the difference between the two values multiplied by the period timer. As depicted in Table 1, generationSNR and distance are very CPU intensive tasks. In this work, we show only the hardware implementation of generationSNR and distance functions. The remainder of the algorithm is

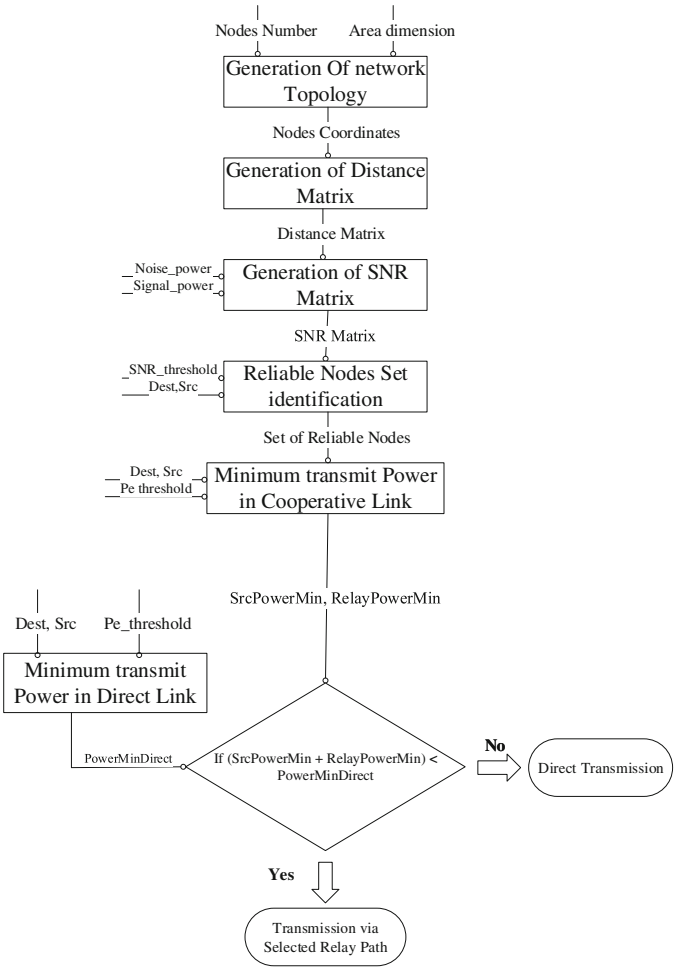


Fig. 3. Relay selection algorithm flowchart

Table 1. Profiling result

Functions	Time (tick)	Time (ms)	Percentage (%)
generationSNR	4 292 435	42.92435	74.118
distance	984 687	9.84687	17.003
optimFunction	480 041	4.80041	8.289
relayReliable	21 854	0.21854	0.377
nodeTopology	8 974	0.08974	0.155
min_distance	1 739	0.01739	0.030
cooperative	1 611	0.01611	0.028

implemented in software. We aim to add accelerators to the standard hardware architecture of the FPGA.

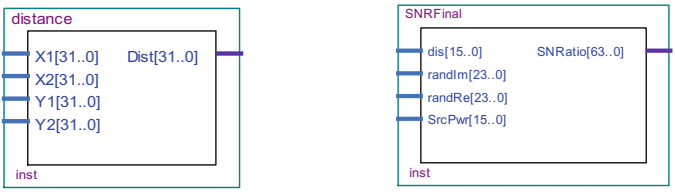
3 Design and Interfacing of Relay Selection Algorithm

In this section, we extend our study to the second level of adaptation which is the architectural level and we study the influence of the architectural attributes on the execution time and energy consumption of the algorithm. We detail the setting up of the accelerators, then we study the impact of the use of these accelerators on the execution time and energy consumption. Various types of System on Chip (SoC) design environment exist. In our work, we choose the one provided by Altera Inc. Our work focuses on the incorporation of hardware accelerators components. Therefore, the system will switch between them according to its needs when executing the software code. The design environment HW/SW consists of a set of software:

- Quartus: development environment for the HW design;
- SOPC Builder: design the hardware part by assembling Intellectual Property (IP);
- Nios IDE: implementation and execution of the software part by emulation or on the map.

3.1 Hardware Accelerators Implementation

To implement the design of HW accelerators, it is necessary to pass through a low-level programming language such as VHDL, Verilog or SystemC. The Altera design environment provides a utility that allows to make blocks in a graphical way through the assembly of predefined operators provided with the environment. It also gives the ability to add blocks written in low-level language. After making the appropriate schematic circuitry, this environment can directly generate the corresponding file into low-level VHDL or Verilog language. After making the entire circuit, we can group all the functions used in a single circuit as a black box that only shows the input/output circuit and for encapsulation treatment, as demonstrated in Fig. 4a and b.



(a) Distance schema block

(b) SNRgeneration schema block

Fig. 4. Distance and SNRgeneration schema blocks

The proper operation of the realized components is verified using simulation. This step is done through test vectors' waveform vector that allow to send values to the inputs of the accelerator and retrieve the results provided. The simulation results show the validation of the proper operation of the SNRgeneration and distance accelerators.

3.2 Test-Bed Results

This subsection is devoted to validate the proposed adaptation model on a reconfigurable platform based on a prototyping ALTERA FPGA (Stratix III) which includes NIOS II as a processor soft core). The development environment are QUARTUS software as development environment for the HW design and NIOS IDE for software implementation, as explained in Sect. 3.1. This development tool allows us to build a reconfigurable architecture around the NIOS processor with specific HW functions implemented as external accelerators via the NIOS Avalon extension bus. In this paper, the accelerators are interfacing through slave mode. To do so, the interface consists of four modules. NIOS-II processor with its memory, the Avalon bus and the accelerators, as shown in Fig. 5. The processor is connected to the Avalon bus and two slave hardware accelerators. Each hardware accelerator specifications are limited to a port of reading and writing. Thus, the master processor should supply the hardware module with data and wait for the result of the slave port. To measure the impact of HW accelerators into the time execution, we use again profiling technique based on software timer and we compare the execution time for different architectures which are:

1. standard architecture with distance HW accelerator;
2. standard architecture with SNRgeneration HW accelerator;
3. standard architecture with both distance and SNRgeneration HW accelerators.

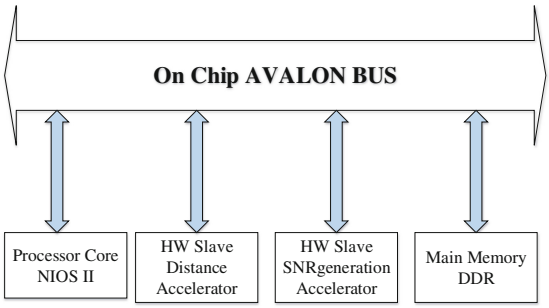


Fig. 5. Interfacing hardware accelerators

Table 2. Profiling result of proposed HW accelerators

Functions	Time (tick)	Time (ms)	Gain (%)
SNR accelerator	1 080 868	10.80868	68.73
Distance accelerator	480 758	4.80758	51.18
SNR+distance (Software)	2 993 399	29.93399	–
2 accelerators	1 342 123	13.42123	55.16

As shown in Table 2, we notice that the use of HW accelerator reduces the number of ticks, thereby decreasing energy consumption. A typical system architecture which includes specific hardware accelerators can improve system performance in terms of execution time. Without loss of generality, we assume that these components have a negligible static power consumption (i.e. if they are not in use, they are placed in high impedance). To measure the impact of the hardware accelerator on the power consumption of the algorithm, we use stratix III card which is incorporated with the necessary circuits in order to display the energy consumption of the FPGA core (mW) during system operation. This method is faster and more accurate but it requires a very specific material with high precision. Therefore, the target algorithm is executed on each type of 3 architectures (listed in the previous paragraph to have new profiling results) to measure the actual consumption to the performance of the algorithm. Table 3 shows the different Hardware configurations consumption measurements. We notice a significant power consumption saving when using hardware accelerators. In fact, we can attain more than 50 % of energy saving when using HW implementation.

Table 3. Energy consumption (J)

Functions	SW implement	HW implement
Distance	8.566	4.278
SNRgeneration	26.042	9.814
SNRgeneration+distance	37.344	12.575

4 Conclusions

In this paper, we opted for a method of HW/SW implementation of a new algorithm based on BEP of selective digital relaying schemes whose purpose is energy consumption reduction. Based on the use of the results provided by profiling and complexity analysis of the latter technique, we identify the critical functions that are CPU intensive tasks and should be implemented as hardware

components. Moreover, we implement and interface the algorithm on a SoC with hardware accelerators to reduce the energy consumption of the sensor node. The HW/SW implementation is tested on FPGA board. Test-bed results showed a significant decrease of the execution time and energy consumption compared with the purely software execution, thereby further prolonging the node as well as the network lifetime.

As future works, we aim to implement an evolvable system on a low power and dynamic reconfigurable FPGA included in a Wireless Sensor Network node.

Acknowledgments. This project is supported in part by the National Electronics, Communications and Photonics Research Center of King Abdulaziz City for Science and Technology (KACST).

References

1. Healy, M., Newe, T., Lewis, E.: Wireless sensor node hardware: a review. In: IEEE Sensors, pp. 621–624 (2008)
2. Gamez, N., Romero, D., Fuentes, L., Rouvoy, R., Duchien, L.: Constraint-based Self-adaptation of wireless sensor networks. In: 2nd International Workshop on Adaptive Services for the Future Internet and 6th International Workshop on Web APIs and Service Mashups, pp. 20–27. ACM (2012)
3. Aziz, S.M., Pham, D.M.: Energy efficient image transmission in wireless multimedia sensor networks. IEEE Commun. Lett. **17**(6), 1084–1087 (2013)
4. Liu, F., Jia, Z., Li, Y.: A novel partial dynamic reconfiguration image sensor node for wireless multimedia sensor networks. In: 2012 IEEE 14th International Conference on High Performance Computing and Communication & 2012 IEEE 9th International Conference on Embedded Software and Systems (HPCC-ICESS), pp. 1368–1374 (2012)
5. Berder, O., Sentieys, O.: Powwow: power optimized hardware/software framework for wireless motes. In: Workshop on Ultra-Low Power Sensor Networks (WUPS), co-located with ARCS, pp. 229–233 (2010)
6. Krasteva, Y., Portilla, J., de la Torre, E., Riesgo, T.: Embedded runtime reconfigurable nodes for wireless sensor networks applications. J. IEEE Sens. **11**(9), 1800–1810 (2011)
7. De La Piedra, A., Braeken, A., Touhafi, A.: A performance comparison study of ECC and AES in commercial and research sensor nodes. In: IEEE EUROCON, pp. 347–354 (2013)
8. Liu, W., Wang, Z., Qu, S., Luo, R.: Reconfigurable network accelerator for wireless sensor nodes. In: 17th International Conference on Advanced Communication Technology (ICACT), pp. 138–142 (2015)
9. Hakim, H., Boujemaa, H., Ajib, W.: Single relay selection schemes for broadcast networks. IEEE Trans. Wirel. Commun. **12**(6), 2646–2657 (2013)
10. Rumberg, B., Graham, D.W., Clites, S., Kelly, B.M., Navidi, M.M., Dilello, A., Kulathumani, V.: RAMP: accelerating wireless sensor hardware design with a reconfigurable analog/mixed-signal platform. In: 14th International Conference on Information Processing in Sensor Networks, pp. 47–58. ACM (2015)
11. Atitallah, N., Hakim, H., Loukil, K., Obeid, A.M., Abid, M.: Energy efficient adaptive transmission strategy using cooperative diversity for wireless sensor networks. In: 27th IEEE PIMRC Mobile and Wireless, pp. 2120–2125 (2016)

Understand Me if You Can! Global Soft Biometrics Recognition from Social Visual Data

Onsa Lazzez(✉), Wael Ouarda(✉), and Adel M. Alimi(✉)

REGIM-Laboratory: Research Groups in Intelligent Machines,
University of Sfax National School of Engineers (ENIS),
BP 1173, 3038 Sfax, Tunisia
{onsa.lazzez.tn, wael.ouarda, adel.alimi}@ieee.org

Abstract. In a relatively short period of time, we have observed the explosion of social network platforms which have acquired a prominent role in the people's daily life. Hence, the extensive use of social networks has generated huge amounts of both visual and textual data that started to gain greater attention. However, the development of effective techniques for the acquisition and analysis of social data has drawn the attention of many researchers. The result of the huge mass of social data from different sources and types has provided many opportunities for researchers in the fields of discovering hidden soft biometrics information from data, which can be used for a variety of applications. In this paper, we propose a novel framework to understand and manage both textual and visual data form social networks, such as Facebook to extract the user' soft biometrics information from posted pictures, specifically age, gender, race and smile.

Keywords: Soft biometrics · Age · Gender · Race · Smile prediction · Social visual data

1 Introduction

In a relatively short period of time, Soft Biometry has drawn the attention of many researchers as a method used to enhance performance of mono-modal biometric system without need to multi-modal architecture. In this strong form, soft biometrics are discrete characteristics that divide people into non-overlapping groups based on soft traits like gender, age, race, weight, height, etc. Soft biometric is a set of features get many biometric information. This information like skin color, height, hair color, gender and age are used by divers existing biometrics systems during enrollment. In scientific literature, the most of biometric systems use a single trait for recognition, for this reason they are also called mono-modal biometric systems. In fact, soft biometrics information can be used to give some sign about user identity or to verify user identity. However, we focus of such soft biometric information traits contained in the visual data include humans' face. In our days, visua l data are generated, an increasing manner, in social networks sites such as Facebook. Social networks have played a significantly increasing role in the peoples' daily life over the last few years. The pervasive use of a

A. Abraham et al. (eds.), *Proceedings of the 16th International Conference on Hybrid Intelligent Systems (HIS 2016)*, Advances in Intelligent Systems and Computing 552,
DOI 10.1007/978-3-319-52941-7_52

social network has generated huge amount of social visual data that started to gain in increasing amount of attention. According to [1], 2016, Facebook users have shared over 4 billion images and 5 billion content, Flicker users have shared over 3 billion images and Google+ users have shared over 1 billion images. In fact, social network sites, such as Facebook, Google+ and Flickr have provided various platforms for a user to share various types of data. Some social networks have platforms that supports textual data, while others have platforms that supports visual data or both textual and visual data. For example, Twitter is used to transmit short text messages, Flickr is used to convey pictures and videos while Facebook and Google+ use all of these data types. However, the rapidly increasing amount of social visual data has led to wide research efforts in recent years [2]. Combining textual and visual data provided by social users reveal an interesting tool to discover unknown soft biometrics information about users. The user's profile picture represents an evident choice to define the user' soft biometrics information. In this dissertation, we investigate the problem of how to use social network data. We tackle the challenge of social users' analysis of the idea of conceiving unknown soft biometrics information about the users. The objective of this paper is to recognize this soft biometrics information, involves, age, gender, race and smile emotional state behind user profile pictures in social network Facebook. The main purpose of this paper include a novel framework named SmartCityZen which aims to collect all textual and visual data available on Facebook users' accounts, a novel dataset named Sm@rtCityZenDB based on our framework SmartCityZen which contains a users' profile pictures and a novel method to recognize Soft biometrics information from users' profile pictures such as gender, age, race and smile emotional state. This paper is structured as follows. Section 2 illustrates various materials and methods about social data and social user's analyzing process. Section 3 details our framework and its different modules. Section 4 shows the attained results. Finally, Sect. 5 recapitulates the paper and presents our outline future works.

2 Materials and Methods

2.1 Social Data

According to [3], social data such the user's name, age, gender, e-mail address, interests, hobbies, favorite sport team(s), athelete(s) or music are essentially provided by users in the social network sites, such as Facebook, Google+, Twitter, etc. on their profiles. After having illustrated the concept of social data, we will present the social user's analysis process in order to create a social database for further analysis.

2.2 Social User's Analysis

In recent years, there has been a huge expansion of the user's provided data in social network sites, such as Facebook and Google+ [4]. In this section, we provide a comprehensive study on recent research on social data analysis. In particular, we focus on the social user's analysis from three aspects: (i) social user's analysis based on textual data, (ii) social user's analysis based on visual data and (iii) social user's

analysis based on both visual and textual data. Table 1 illustrates the recent works on the social users' analysis process.

2.2.1 Social User's Analysis Based on Textual Data

Analyzing the textual data on the social network sites are a dynamic research topic in the current years. For example, as reported by [5], the authors proposed an approach to analyze the users' tweets in order to classify their feelings. In [6, 7], the authors developed a novel framework that aims to identify topical experts of terrorism in Twitter based on the users' generated textual data. However, textual analysis has proved to be quite powerful, but not perfect and suffers from the need to develop language specific models for different cultures. As social visual data tend to be various, conversational and rapid evolving in content, the use of only textual data is too limited.

2.2.2 Social User's Analysis Based on Visual Data

Visual data in social network sites are mainly presented by images. Hence, visual data analysis can be adapted for many research areas, such as the users' feelings analysis, the users' attribute prediction, etc. [8]. In [9], the authors proposed a novel idea about automatically capturing the users' feelings expressed in selfies provided by users in social networks. According to [10, 11], the authors proposed a novel approach to collect visual data, particularly images from Twitter to predict the user's gender by analyzing the colors adopted by the users in their pictures. In fact, these approaches are limited to the analysis of only visual data. Hence, textual and visual data are complementary and their combination increases the users' attribute prediction precision and feelings analysis process.

2.2.3 Social User's Analysis Based on Both Textual and Visual Data

In the social users' analysis process, the fusion between textual and visual data provides a more efficient and true results. The works in [2, 12] present a microblog filtering method which can filter out noisy data from the collected microblogs for a particular brand. A brand filtering framework was proposed to draw the social visual data, microblogs, into a latent subspace that is not only based with respect the target brand but also regular with various types of information comprising textual and visual data [13]. It should be noted however that, these works have considered both textual and visual data from microblog platforms to gather brand-related data. The main challenges of these works that it is more important to utilize other social networks, such as Facebook, Google+, etc. enable to gather more and more mass of visual and textual data unlike the microblogs, in the fact which enable to collect, only short texts, and embedded images.

Table 1. Social user analyzes

Data type	Problem	Data unit
Textual	Classify users' feelings [5]	Tweets
	Identify topical expert of terrorism [6]	
Visual	Visual feeling analysis [9]	Selfies
	Gender prediction [10]	Image content
Textual and visual	Filter data for a particular brand [2]	Image content and context

All works already mentioned focus mainly on Twitter in the social users' analysis process by taking into account the textual and visual data or both. In our work, we will focus on Facebook social networks to collect data from the users' profiles and then analyze visual data to predict the users' soft biometrics information such as age, gender, race and smile.

3 Realized Work

An important observation of social networks is that they contain a rich level of information about the users: posts, comments, images and favorites. The visual data accumulated from social networks, like Facebook, Google+, Twitter, etc. aim at enhancing the social user's analysis process to discover hidden information about users. In this context, we propose a novel system for social user's analysis, which includes the following components: SmartCityZen framework, SmartCityZenDB dataset and Face++ system. The Fig. 1 illustrates our proposed system. We will detail in the following sections, each step/module in our proposed system.

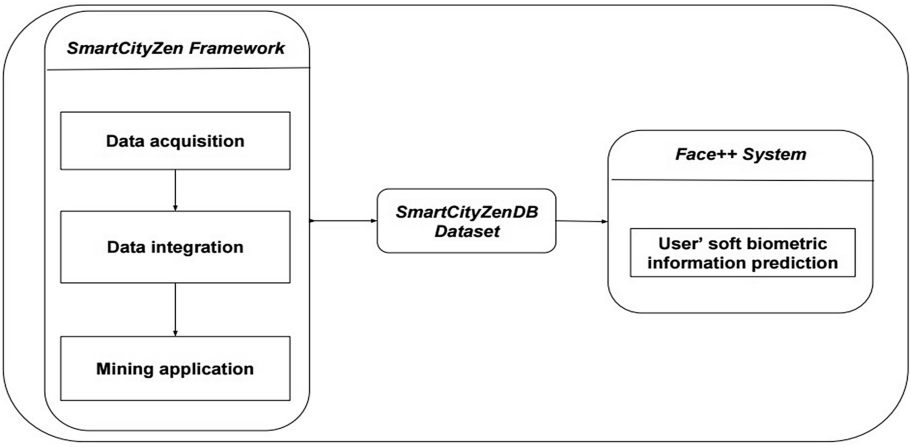


Fig. 1. General architecture of proposed system

3.1 SmartCityZen Framework

Our framework contains essentially 3 basic modules: (1) Data Acquisition, (2) Data Integration and (3) Mining Application. As shown in Fig. 2, the main focus is the data acquisition module is how to collect all the textual and visual data available in Facebook and Google+ users' profiles. Secondly, in the data integration model, we will store different types of data which are accessible on multiple users' profiles, such as posts, comments on posts, shares of posts, etc. Lastly, mining applications are utilized as an example of soft biometrics information prediction: Users' gender, age, race and smile prediction.

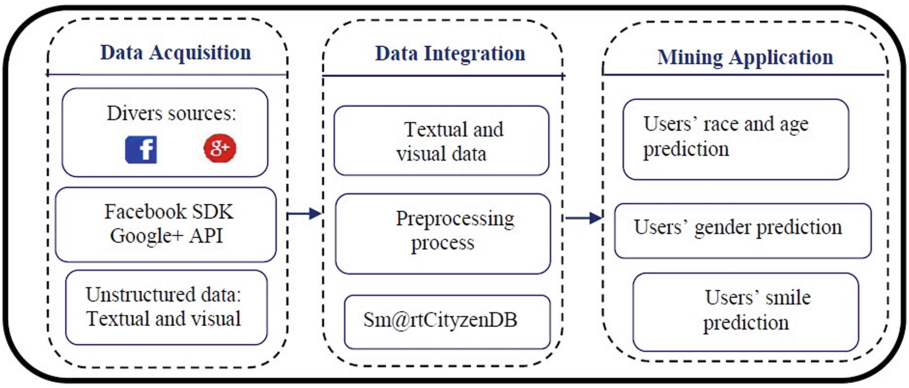


Fig. 2. SmartCityZen framework architecture

3.1.1 Data Acquisition

Facebook is the most important among all the social networks due to its largest user population. Moreover, it has the most active users with around 800 million in 2016 [11]. On its part, Google+, which is relatively a new social network, enables users to share data with specific groups within their personal network. Therefore, social data acquisition from various social network sites is an important and hard task, which has attracted the attention of many researchers [2].

3.1.2 Data Integration

Combining the characteristics from visual and textual data generated by the user reveals curious proprieties of social user analysis and serves as a powerful method of discovering the hidden information about the users. In this context, a database was created to store all the data obtained from Facebook and Google+ users' profiles. The database named Sm@rtCityzenDB was developed using PHPMYadmin. Its structure is depicted in Fig. 3. For an evaluation surpose, we collectes about 50 Facebook and Google+ users' accounts.

3.1.3 Mining Application

In this section, we introduce our system to estimate the users' age, gender and smile from visual data, using profile pictures. Figure 4 shows our system architecture to predict the users' soft biometrics information such as age, gender and smile.

3.2 SmartCityZenDB Dataset

We have constructed a novel dataset named SmartCityZenDB which contains 50 users' picture profiles collected by our framework SmartCityZen. Figure 5 presents our database.

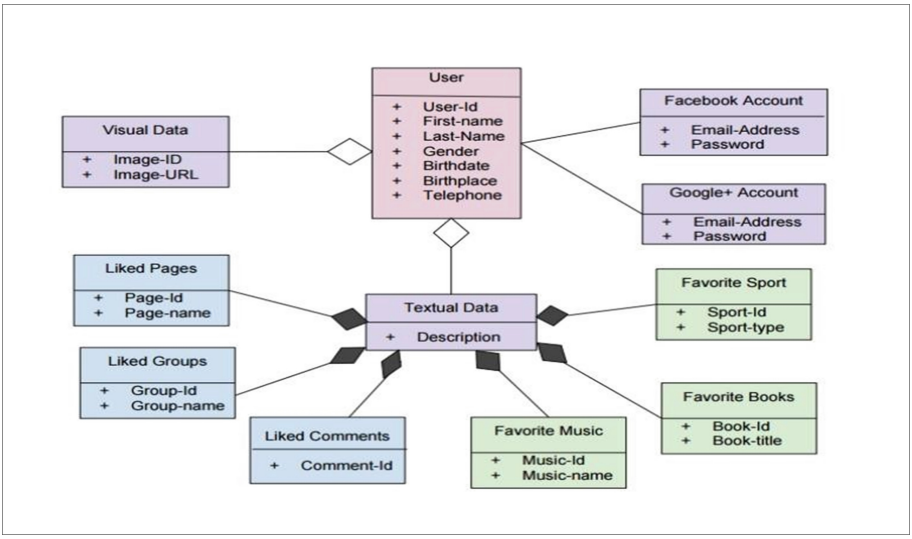


Fig. 3. SmartCityZenDB dataset structure

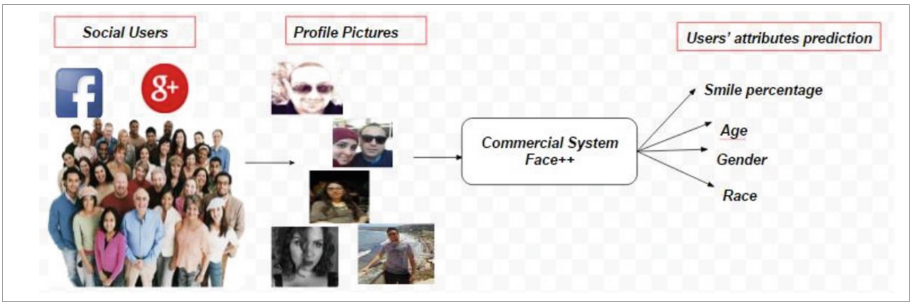


Fig. 4. Proposed age, gender, race and smile prediction system

3.3 Face++ System: Face-Based Age/Gender/Race and Smile Predictor

In order to estimate global soft biometric information of the users based on their profile pictures, we performed the Face++ System [14], which includes many modules like face detection, features extraction based on landmarks and demographic information classification like gender, age, race and the percentage of smiling in the user profile picture. This Api exclude automatically pictures without face. For each input image collected automatically from our developed Sm@rtCityzen platform, the system returns detected face with user attributes mentioned (gender, age, race and smiling emotional state). We should notice that to estimate soft information attributes, the system uses 83 facial points including all facial parts like face edge, mouth, nose, eyes and eyebrows. All trained image are around 5171 faces taken in the wild [15].

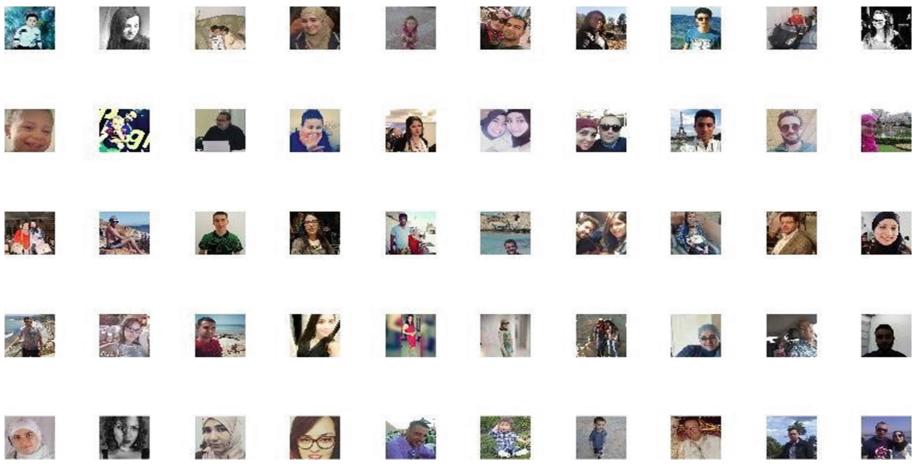


Fig. 5. Some profile pictures collected to construct our Sm@rtCityzenDB

4 Experiment Results

4.1 Qualitative Results for Visual and Textual Social Data

Our system was tested on new constructed database containing 50 subjects. The constructed database includes different persons with several ages, gender, race and emotional states. Each person connects to our platform, Fig. 6, using his Facebook or Google+ account. Our Sm@rtCityzen platform collect automatically textual information as given in Figs. 7, 8, 9 and 10 and Visual information as shown in Fig. 11.

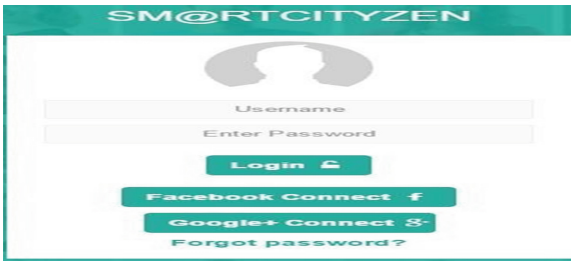


Fig. 6. Sm@rtCityZen framework connexion

Figure 7 illustrates the basic textual data about user such as picture profile, mail address, phone number, bio, etc.

Beyond the basic information, we collected all the favorite books, music and sports which contain a textual description for each book, music and sport.

For visual data, Fig. 11 illustrates social visual data as picture shared in user' profile.

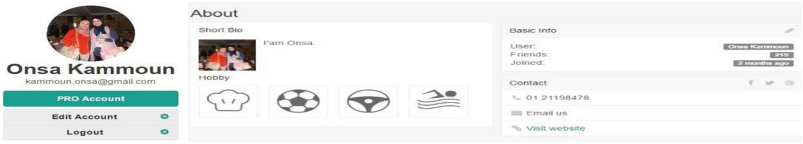


Fig. 7. Basic information from the user's profile collected using Sm@rtCityzen platform

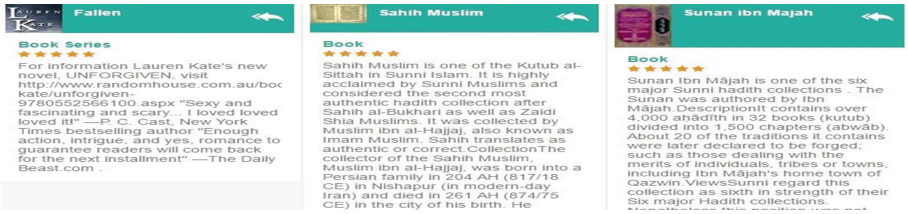


Fig. 8. User' favorite books collected using Sm@rtCityzen platform

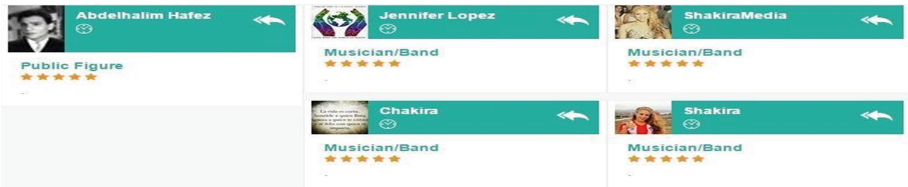


Fig. 9. User' favorite music collected using Sm@rtCityzen platfom

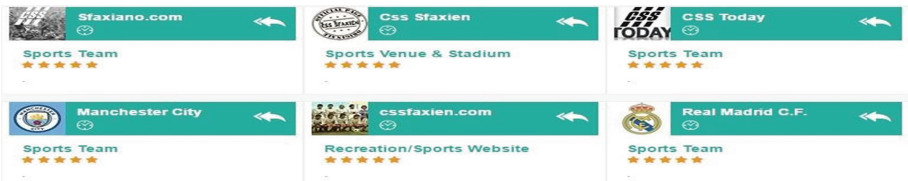


Fig. 10. User' favorite sport collected using Sm@rtCityzen platform

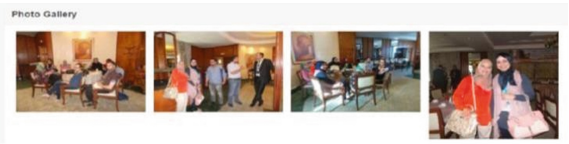


Fig. 11. Visual data collected using Sm@rtCityzen platform

4.2 Qualitative Soft Biometrics Information

From the user' profile picture, Face++ estimates gender, age, race and smiling as given in Fig. 12 which presents a female which has an age in the range of 21 and 31 years, a white race and a 24,75 smiling percentage.

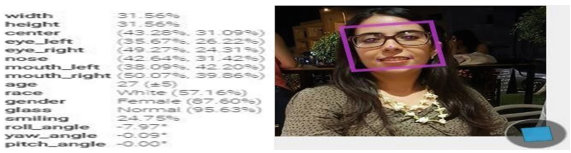


Fig. 12. The user' soft biometrics information prediction

4.3 Quantitative Results

We tried, through our Sm@rtCityzen framework, to predict the citizens' Soft Biometric information from their profile pictures to showcase the visual data importance in Facebook. Indeed, we have estimate the age, gender and smile of 50 citizens' accounts participants in our Sm@rtCityzen platform and stored in our Sm@rtCityzenDB. Table 2 illustrates the accuracy of our dataset Sm@rtCityzenDB. Figure 13 illustrates our estimation results obtained.

Table 2. Accuracy of our SmartCityZenDB

Soft biometrics	Accuracy
Age	96%
Gender	98%
Race	94%
Smile	100%

In order to analyses our database Sm@rtCityzenDB, we can interpret the list of the figures presented above. Firstly, by the Fig. 12(a), we deduct an equal proportion of women and men. Secondly, from the Fig. 12(b), we note that the majority of citizens have an age between 20 and 30, one citizen with an age higher to 60 and some citizens with an age lower to 10. Furthermore, in Fig. 12(c), we observe the existence of various types of race as White, Black and Asian. Finally, in term of smiling, in Fig. 12(d), we observe that the majority of citizens have a smiling percentage higher of 40%. From this interpretation, we can deduct that our database named Sm@rtCityzenDB is very diverse in terms of age, gender, race and feelings. In fact, it contains an equal number of woman and men, several age categories as baby, young, adolescent, adult and seniors, several races and diverse feelings. Based on our experiments, we should make following observations: The user's soft biometrics information classification gives an interesting rates 25, Female, white and 45% respectively for age, gender, race and smiling.

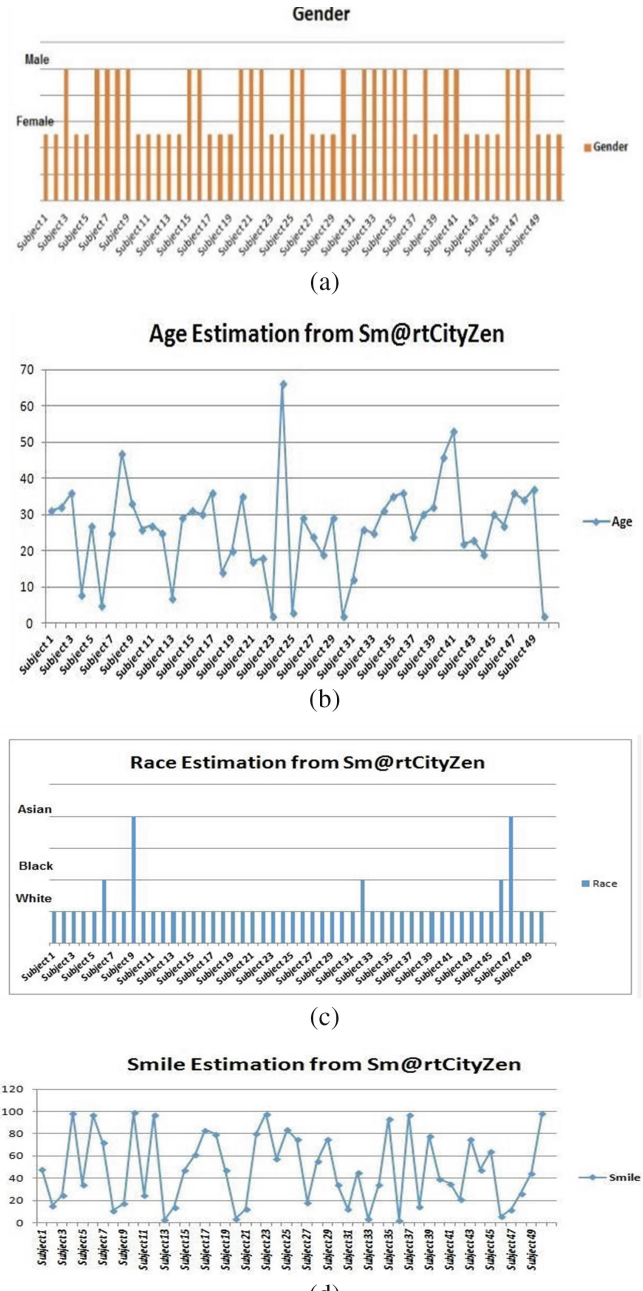


Fig. 13. Citizen's' attributes estimation from Sm@rtCityZen: (a) Gender estimation, (b) Age estimation, (c) Race estimation and (d) Smile emotional estimation

Our constructed database presents the following percentages of user profile:

- 48% Male; 52% Female.
- 20% under 10; 16% between 10 and 20; 38% between 20 and 30 and 26% upper than 30.

The smiling emotional state is not always the case of profile picture user as shown in Fig. 12(d). The average smiling rate is 45% and the mean age is 25 of users' profiles collected in our Sm@rtCityzenDB.

5 Conclusion

Information on the internet is growing exponentially and in recent years, social networks are being widely used. In this context, we have utilized the social network data by creating a novel database named Sm@rtCityzenDB which contains 50 citizens' account contains all visual and textual data from users' profiles for discovering soft biometrics information about users such as age, gender and smile from their pictures profiles and their basic information. To create our database, we have developed a novel platform named Sm@rtCityzen. In fact, our platform is accessible by diverse citizens. Indeed, our database Sm@rtCityzenDB contains various citizens in terms of age, gender, race and smiling percentages. This developed platform will be the basis of our future work. We aim as extension to integrate our Sm@rtCityZen platform in order to perform our new personalized search engine based on user profile information like demographic soft biometrics (gender, age, and race) and preferences.

References

1. Chaffty, D.: Global social media research summary 2016. National Archives, Office of Innovation, Social Media Team (2016)
2. Gao, Y., Zhen, Y., Li, H.: Filtering of brand-related microblogs using social-smooth multiview embedding. IEEE Trans. Multimedia **18**, 2115–2126 (2016)
3. Batool, R., Khattak, A.M., Maqbool, J., Lee, S.: Precise tweet classification and sentiment analysis. In: IEEE/ACIS 12 th International Conference on Computer and Information Science (ICIS) (2013)
4. Bouhnin, C., Géry, M., Legon, C.: Personalized information retrieval models integrating the user's profile. In: Research Challenges in Information Science (RCIS) (2016)
5. Kaur, K.: Development of a framework for analyzing terrorism actions via Twitter lists. In: IEEE International Conference on Computational Techniques in Information and Communication Technologies (ICCTICT) (2016)
6. Tedmori, S., Al-Lahaseh, R.: Towards a selfie social network with automatically generated sentiment-bearing hashtags. In: 7th IEEE International Conference on Computer Science and Information technology (2016)
7. Imran, M., Elbassuoni, S., Castillo, C., Diaz, F., Meier, P.: Practical extraction of disaster-relevant information from social media. In: Proceedings of International ACM World Wide Web Conference Committee (IW3C2), Rio de Janeiro, Brazil (2013)

8. Ghosh, S., Sharma, N., Benevenuto, F., Ganguly, N., Gummadi, KP.: Cognos: crowdsourcing search for topic experts in microblogs. In: Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval, New York, USA (2012)
9. Merler, M., Cao, L., Smith, J.R.: You are what you tweet…PIC! Gender Prediction based on semantic analysis of social media Images. In: IEEE International Conference on Multimedia and Expo (ICME) (2015)
10. Bonneau, J., Anderson, J., Danezis, G.: Prying data out of a social network. In: Advances in Social Network Analysis and Mining (2009)
11. Borth, D., Ji, R., Chen, T., Breuel, T., Chang, S.-F.: Large-scale visual sentiment ontology and detectars using adjective noun pairs. In: Proceedings of the 21st ACM International Conference on Multimedia (2013)
12. Ntalianis, K., Doulamis, N.: An automatic eventcomplementing human life summarization scheme based on a social computing method over social media content. In: Multimedia Tools and Applications (2015)
13. Ma, X., Tsuboshita, Y., Kato, N.: Gender estimation for SNS user profiling using automatic image annotation. In: ICME Workshop on Cross-Media Analysis from Social Multimedia (CASM) (2014)
14. Fan, H., Yang, M., Cao, Z., Jiang, Y., Yin, Q.: Learning compact face representation: packing a face into an int32. In: ACM Multimedia (2014)
15. Zhou, E., Fan, H., Cao, Z., Yin, Q.: Extensive facial landmark localization with coarse-to-fine convolutional neural network. In: ICCV Workshop (2013)

Using Data Clustering on ssFPA/DE- a Search Strategy Flower Pollination Algorithm with Differential Evolution

Meera Ramadas[1(✉)], Ajith Abraham[2], and Sushil Kumar[3]

[1] AIIT, Amity University, Noida, Uttar Pradesh, India
meera_mgr@rediffmail.com
[2] MIR Labs, Auburn, USA
[3] ASET, Amity University, Noida, India

Abstract. Evolutionary algorithm exist as a subclass of artificial intelligence that requires constant optimization. It is an area of immense interest to several researchers. Different biological behaviors has formed the base for implementation of various algorithms like firefly, genetic, bee colony particle swarm optimization. Flower Pollination Algorithm (FPA) is the most recent work in this field, which use the flower pollination technique. Differential Evolution (DE) is a basic and powerful computation showing a strong global optimization. This article gives a brief about FPA and DE. Subsequently, a hybrid algorithm named as ssFPA/DE using the search strategy of flower pollination algorithm in differential evolution is applied in the field of clustering for checking the efficiency of the algorithm.

Keywords: Optimization · Levy · Mutation · Crossover · Probability · Generation · Selection · Clustering

1 Introduction

For centuries, solution to several problems were derived based on various biological system that exist in nature. Many of them depicts improved efficiency in maximization of numerous evolutionary process. Based on these biological behaviors, algorithms like the genetic, firefly and particle swarm optimization algorithms were introduced. Evolutionary algorithm demands steady optimization. Genetic algorithm forms a component of the larger set of evolutionary algorithm that solve several optimization problems by replicating the techniques that exist in nature like selection, crossover, inheritance and mutation. Genetic algorithm follow Darwinian theory of natural evolution. In 1997, Price and Storn tried to alter the traditional crossover and mutation operators in genetic algorithm with another operant and subsequently introducing a differential operator for managing the situation. The proposed algorithm being referred as Differential Evolution (DE). DE is a technique that improves a problem by continuously trying to upgrade an entry with respect to a given measure of quality. The properties of DE such as ease of use, robustness, compact structure and speed has made the researchers to continue with their work in improving the DE efficiency.

A. Abraham et al. (eds.), *Proceedings of the 16th International Conference on Hybrid Intelligent Systems (HIS 2016)*, Advances in Intelligent Systems and Computing 552,
DOI 10.1007/978-3-319-52941-7_53

FPA is a recent work in this field of evolutionary algorithm, which use the flower pollination technique to solve constrained and unconstrained problem. FPA has the property of quick execution and ease of modification.

Analysis of large amount of data in day-to-day work has become inevitable in many professions. With the vast amount of information currently available, the necessity to classify and cluster such data has become necessity. Various techniques for clustering has been developed even using various evolutionary algorithms. Here, an attempt has been made to implement the hybrid algorithm developed with DE and FPA to clustering technique.

2 Differential Evolution: Background and Literature

Storn and Price [2] proposed DE, which was a population, based random search technique that showed decent performance over a number of problems. Based on diversity between two solutions termed as vectors, the DE make use of a mutation operator to decide the search direction. The direction of search is decided depending on the scattering of results in the group. DE engages balanced state switching technique where a newly developed offspring known as trial vector clash with its comparable parent. The trial vector displaces parent if new offspring possess a desirable fitness value. EA perform the recurring process of growth and development in population. DE is among best of techniques used to compute the real valued test function that obtain the overall best of a function within continuous space.

A specified number of vectors are arbitrarily identified which are derived over time in a search space of n-dimensions of likely solutions. The local minima for the objective functions are derived. In each generation or iteration, a new vector is formed by combining two or more vectors been randomly identified within the population. This process is referred as mutation. The outcome vector is with predetermined target vector. In the process called recombination, if a trial vector produce a better value of the objective function, then it will be accepted for the next generation. This process is called as selection. Unless some ending criteria is fulfilled, the mutation, recombination and selection shall continue. DE uses a population of NP candidate solutions denoted as $X_{i,G}$ where $i = 1, 2 \ldots$NP where index i denote population and G represents generation of population. Differential Evolution algorithm depends on the three operations mainly mutation, selection and reproduction.

Mutation: This operator causes DE to be distinct from other Evolutionary algorithms. It calculates the weighted difference in-between the vectors in population. Mutation starts by arbitrarily choosing three individuals from the population. This operation extends the workspace. For the given parameter, $X_{i,G}$ we are arbitrarily selecting 3 vectors $X_{r1,G}, X_{r2,G}$ and $X_{r3,G}$ such that r_1, r_2, r_3 are distinct. Then the donor vector $V_{i,G}$ is computed as:

$$V_{i,G} = X_{r1,G} + F \times (X_{r2,G} - X_{r3,G}) \tag{1}$$

Here F is the mutation factor which is a constant from [0, 2].

Crossover: This process also termed as recombination, includes successful solutions into the population. The trial vector $U_{i,G}$ is created for the target vector $X_{i,G}$ using binomial crossover.

The Components of the donor vector enter the trial vector with the probability $C_r \in [0, 1]$. C_r is crossover probability that is selected along with population size $NP \geq 4$.

$$U_{j,i,G+1} = \begin{cases} V_{j,i,G+1} & if\ rand_{i,j}[0,1] \leq C_r\ or\ if\ j = I_{rand} \\ X_{j,i,G+1} & if\ rand_{i,j}[0,1] > C_r\ or\ if\ j \neq I_{rand} \end{cases} \tag{2}$$

Here $rand_{i,j} \approx \cup [0, 1]$ and I_{rand} is random integer from 1,2…N.

Selection: This operation differs from the selection operation of other evolutionary algorithms. Here the population for next generation is chosen from vectors in current population and its subsequent trial vectors. The target vector $X_{i,G}$ is matched with the trial vector $V_{i,G}$ and the least value of function is taken into next generation.

$$X_{i,G+1} = \begin{cases} U_{i,G+1} & if\ f(U_{i,G+1}) \leq f(X_{i,G})\ where\ i = 1, 2, \ldots N \\ X_{i,G} & otherwise \end{cases} \tag{3}$$

Mutation, crossover and selection operations are continued until some stopping criteria is attained.

Das et al. [8] detailed various techniques of differential evolution and explained various areas of work in DE with the listing of its areas of application. DE algorithm can not be applied on linearly separable functions. DE has difficulties on functions that are non-linearly separable or limitation when trying to move its population over large space. There is immense opportunity for research to make DE robust. Probabilistic convergence, martingale theory, drift analysis, stochastic lyapunov energy function of DE are an open issue for research.

3 Flower Pollination Algorithm: Background and Literature

Yang et al. [7] produced a work on flower pollination algorithm for multi-objective situations using the technique of pollination of flowering plants. Pollination in nature is explained as the process of transferring pollens from the male stamen to female stigma of the flower. The intention of any living thing shall to produce off springs for the succeeding genesis. One of the method plant follow is by producing seeds. There are two types of Pollination in plant:

- cross-pollination
- self-pollination

Cross-pollination happen, when the pollens from one of the plants get transferred on to flower of the other plant. This takes place by the means of biotic and abiotic creature like the bird, insect, bee, etc.
Self-pollination occur when the flower pollination happen with-in same plant. FPA which was modeled by Xin She Yang (2012) follows four rules.

- Local pollination consist of abiotic and self pollination methods
- Global pollination consist of biotic and cross pollination methods
- Likeness of two flowers considered is directly related to the duplication probability
- The global and local pollination is controlled by Switch probability

These rules are converted to equation as given below:

$$x_i^{t+1} = x_i^t + L(x_i^t - g_*) \tag{4}$$

$$x_i^{t+1} = x_i^t + \varepsilon(x_i^t - x_{ki}^t) \tag{5}$$

Where x_i^t denote solution vector, g_* the current best during iterations, L is the step size from Levy distribution and ε is any value from distribution [0,1]. This distribution is a continuous probability distribution for positive random numbers. Flower Pollination algorithm is flexible and exponentially better to solve with its application mainly in the field of engineering and fuzzy logic.

4 Clustering Technique: Background and Literature

With the advances in technology, the need for acquiring, storing and processing a huge amount of data is ever increasing. Clustering is a method of splitting a group of objects into a set of meaningful sub divisions or classes called clusters. It improves in understanding the natural grouping in a data set. It is breaking down of large population into smaller groups that are similar in character. A cluster is a pool of data which are alike internally, but dissimilar to objects in other clusters. Dissimilarities and similarities are evaluated based on properties of attribute defining the objects. It is a connected region of a multidimensional space containing a relatively high density. The superiority of a clustering result is also based on the similarity measure applied by the method and its implementation. The standard of a clustering method is determined by its skill to find some or all of the hidden patterns. High intra cluster likeness and low inter cluster likeness are considered while grouping data into clusters. While performing the cluster analysis, first partition data into groups on the basis of data similarity. Then we allocate the label to the groups. The benefit of clustering over classification is that it is flexible to modifications and aids to distinguish useful features that identifies different groups. Clustering is also called as unsupervised learning as labelled documents are not understood in clustering. Document clustering is used extensively in fields of web mining, search engines, topological analysis and information retrieval (Fig. 1).

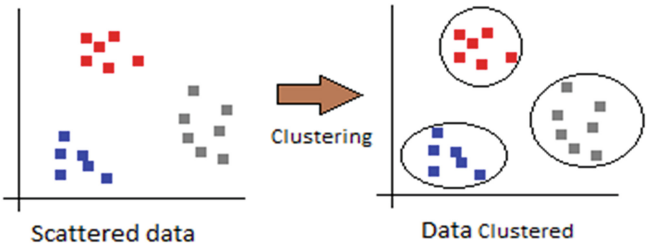


Fig. 1. Scattered and clustered data

Clustering is built on three features: exclusiveness, nesting and completeness. In nested type, division is based on the features of nesting clusters. Hierarchical clustering are nested clusters. In hierarchical type, a set of nested clusters are arranged as a hierarchical tree. Exclusive separation is built on the features that allows a data object to occur in one or more than one clusters. Here each data object can only exist in one cluster. Completeness is a type of distinction based on the features that requires all data objects to be grouped. A complete clustering allocates every object to a cluster. Various steps are involved in data clustering like data collection, initial screening, representation, clustering tendency, clustering strategy, validation and interpretation. Data collection involves the removal of relevant data from the data sources. Initial screening is the messaging of data after extraction from source. Representation is the preparation of data to make it suitable for the algorithm. Clustering tendency verifies the data in hand for its usage in cluster or not. Clustering strategy chooses the correct algorithm and parameter. Validation is the manual examination of the data for validity of technique. Interpretation includes combining clustering results with other studies and suggest further analysis.

Partitional algorithm permits to divide data into sets of dissimilar clusters using similarity criterion. It allows updating of cluster members if the clustering performance is improved. Partitional clustering is an optimization problem as it minimizes the cluster assignment in probability density function. There are basically two standard type of clustering: partitioning algorithm and hierarchical algorithm. Partitioning algorithm specifies an initial number of groups and iteratively performs the reallocation of objects among groups to convergence. K-means is a noted partitioning algorithm introduced by Lloyd (1957). Objects are placed into k-groups, k chosen a prior. Centroid for each group is calculated and each object is placed into the group based on its distance to centroid. This method reduces the overall within cluster dispersion by iteratively reallocating cluster member. Iterations are done for updating the assignment of data to clusters and for updating the cluster centers. There are two types of clustering: soft clustering and hard clustering. In hard clustering, each data is a member of one cluster. In soft clustering, data assignment are distributed over all clusters.

The aim in K means algorithm is to produce k clusters from a set of objects $x_1, x_2, x_3 \ldots x_n$. Place the centroids $V_1, V_2 \ldots V_n$ at random locations. The distance measure usually employed is the Euclidean distance. It is calculated as:

$$dist(V,x) = \sqrt{\sum_i (V_i - x_i)^2} \tag{6}$$

Where V is the set of center and x is the set of data objects. The cluster center for choosing the best interval for each cluster is calculated as:

$$V_i = \left(\frac{1}{C_i}\right) \sum_{i=1}^{C_i} x_i \tag{7}$$

Where C_i is the number of data points in the i^{th} cluster.
The algorithm is as follows:

1. Cluster data into k clusters.
2. Choose k arbitrary points as cluster centers.
3. Allocate each object to the nearest center by calculating distance using Eq. (6).
4. Recalculate centers of cluster using Eq. (7).
5. Perform steps 2 to 4 until centers do not change.

Wei Lu *et al.* [4] gave a new evolutionary algorithm that applies on an entropy based principle which properly estimated the optimal number of clusters for a dataset. Here two sets of data: synthetic data and a standard Irish dataset are used for algorithm validation. Martínez-Estudillo *et al.* [9] proposed a hybrid evolutionary algorithm to solve non – linear regression problems. The algorithm was the combination of a clustering process, an evolutionary algorithm and a local search procedure where only some individuals were subject to local optimization. Alves *et al.* [1] gave an improved version of evolutionary algorithm for clustering called F-EAC (Fast Evolutionary Algorithm for Clustering). Here, the influence of fitness function was reduced in the assessment process. Zhang *et al.* [5] introduced a genetic algorithm to solve the clustering aggregation problem. Here the aim of this algorithm is to code clustering division where this clustering division is the chromosome of genetic algorithm. Zheng *et al.* [6] suggested an unsupervised evolutionary clustering algorithm for mixed data type named Evolutionary K-Prototype Algorithm (EKP). Here, a comparison is done with the traditional K-Prototype (KP) algorithm. Voges *et al.* [3] proposed an evolutionary based rough clustering algorithm which does not need to indicate in advance the number of clusters and was independent of initial starting point. It gave an overview on rough clusters and on evolutionary algorithm for development of viable cluster solution, which consist of an optimal number of templates providing description of clusters.

5 ssFPA/DE Hybrid Method

A hybrid optimization method was developed using DE and FPA. The initial population is produced using FPA. Here, using the concept of local and global pollination, the different search strategies of local and global type is formed. Subsequently, using

the FPA algorithm, the best solutions for the initial population are generated and treated as new population. These best solutions will further undergo differential evolution and results obtained. Here, the population from FPA will undergo mutation and crossover. This generates the final population.

```
Max or min of objective f(x), x = (x_1, x_2, ...x_d).
Assign n pollen seeds. Choose optimal value (g_*) in the original group
Initialize number of iterations to t and a switching probability to p ∈ [0,1]
 While (population > t )
      For every n  flowers of the group
            If ( p >  rand)
                  choose a step vector L  with d dimension
                  Perform global pollination using equation  4

            else
                  Choose product within a uniform distribution
                    in [0,1]
                  Perform local pollination using equation 5
            end if
            Calculate fresh results
            If fresh results are superior, then re-evaluate them in group
      end for
end while
For each object X  in the set do:
      Perform DE algorithm
End for
End algorithm
```

The algorithm was executed using MATLABr2008b and results obtained for few standard functions of DE, which was then compared with original DE approach.

6 ssFPA/DE in Clustering

The fitness of each solution is performed by evaluating the distance between the centroid and the entity point which is defined as:

$$Fitness(C) = \sum_{j=1}^{k} \sum_{i=1}^{n} \left\| x_i^j - c^j \right\|^2 \tag{8}$$

where x_i^j is the entity point, c^j is the centroid and $\left\| x_i^j - c^j \right\|$ gives the distance between the centroid and the entity point. The algorithm developed is used in data clustering. Here the k- means technique of clustering is used. The proposed algorithm is as below:

Proposed Algorithm for clustering:

```
Cluster data into k groups.
Select arbitrary k points as cluster center.
Allocate objects to their adjacent cluster center by
calculating distance using Euclidean method formula (6)
Find the cluster center for choosing best interval for each
cluster using (7)
Generate initial solution to find best cluster center.
Calculate the cost function for each solution in algorithm
While (termination criteria not reached)
    Sort the solution based on objective function value.
    For (all group)
          For (all sites in group)
                Perform hybrid algorithm
                Calculate cost function for trial site
          End for
    End for
End while
End algorithm
```

7 Experimental Results

The above mentioned hybrid algorithm was implemented using MATLABr2008b running on i7 core processor, 64 bit operating system with 12 GB RAM and a comparative result with the original DE algorithm is obtained. We have used different standard functions and evaluated the results by fixing the value to reach and number of iterations. Few of the results are tabulated in Table 1 where the hybrid algorithm is compared with the classical DE algorithm. It was noted that the proposed hybrid algorithm gave the best value for most of the standard functions.

Based on Best Value:

The tabulated values were graphically plotted. The graphs shows performance curve of few different function strategies. The x-axis depict the number of function evaluation and y-axis depict the objective function. The graph is plotted for various values at each iteration for fixed VTR value of e−015 and dimension size of 25 (Figs. 2 and 3).

Table 1. Best value for different functions

Function	D	DE					Hybrid				
		DE/best/1	DE/rand/1	DE/best-to-rand/1	De/best/2	DE/rand/2	DE/best/1	DE/rand/1	DE/best-to-rand/1	DE/best/2	DE/rand/2
Sphere	25	9.73e−016	6.2e−016	7.532e−016	9.655e−016	7.17e+0	8.62e−016	8.967e−016	9.13e−016	9.325e −016	6.1900e+
	50	8.5e−016	9.99e−016	6.2e−016	**9.7e−016**	2.517e−001	8.06e−016	8.47e−016	7.73e−016	8.84e−016	2.301e−001
Booth	25	3.265e −016	2.3e−016	1.713e−016	7.587e−016	7.725e−016	5.13e−016	3.253e−016	**1.193e−016**	3.307e −016	4.4782e −016
	50	3.17e−016	5.87e−016	1.76e−016	5.9e−016	3.7−016	3.72e−016	**7.2e−017**	1.9e−016	5.68e−016	8.9e−016
Beale	25	3.497e −016	2.05e−016	6.0738e−016	7.0792e −016	8.35e−016	9.3042e −016	8.7327e −016	**1.804e−016**	3.45e−016	2.724e−016
	50	5.9e−016	2.56e−016	3.8e−016	9.73e−016	3.9e−016	8.85e−016	7.61e−016	5.313e−016	6.82e−016	**7.88e−017**
Schwefel	**25**	**−1.8e+003**	**−2.2e+003**	**−7.840e+001**	**−1.38e+003**	**−1.166e +003**	**−8.0e+002**	**−4.63e+002**	**−4.07e+001**	**−6.82e +002**	**−2.808e +002**
	50	**−5.6e+002**	**−2.48e +003**	**−6.4e+002**	**−4.93e+003**	**−2.4e+003**	**−.4e+003**	**−5.04e+002**	**−2.48e+003**	**−5.6e+003**	**−4.43e+003**
Michalewicz	**25**	**−7.69e+00**	**−7.2e+00**	**−7.39e+00**	**−6.959e+00**	**−6.847e+00**	**−6.343e+00**	**−6.89e+00**	**−8.09e+00**	**−6.475e +00**	**−7.036e+00**
	50	**−1.13e +001**	**−1.2e+001**	**−1.21e+001**	**−1.24e+001**	**−1.12e+001**	**−1.04e+001**	**−1.24e+001**	**−1.097e+001**	**−1.08e +001**	**−1.18e+001**

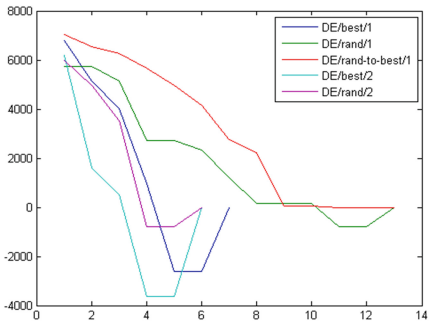


Fig. 2. Schwefel function using DE

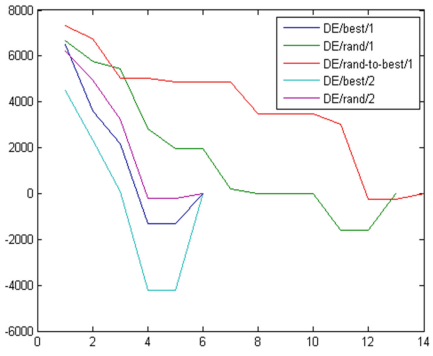


Fig. 3. Schwefel function using ssFPA/DE

Based on the values from Table 1, Friedman test was applied and the results obtained were tabulated. Table 2 represents the values obtained from the test performed on both DE and ssFPA/DE.

Table 2. Test statistics using Friedman's test

N = 25	DE	ssFPA/DE
Chi sq	15.3	14.12
Df	5	5
Asymptotic significance	0	0.004

The proposed hybrid algorithm was applied to clustering for three different artificial datasets. The details are given in Table 3. Results were obtained for the various datasets shown below.

Table 3. Details of dataset

Name of dataset	Number of elements	Dimension
Dataset1	300	2
Dataset2	240	2
Dataset3	110	2

The clustering plot for the various datasets are given below. The data was clustered into 3 and 4 clusters (Figs. 4, 5, 6 and 7).

For the various datasets, the subsequent curve graph for the best cost at each stage of clustering is also plotted (Figs. 8 and 9).

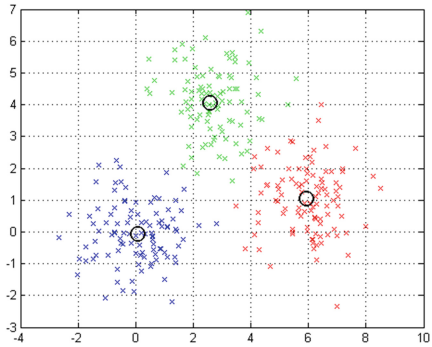


Fig. 4. Cluster plot for Dataset1 for DE algorithm

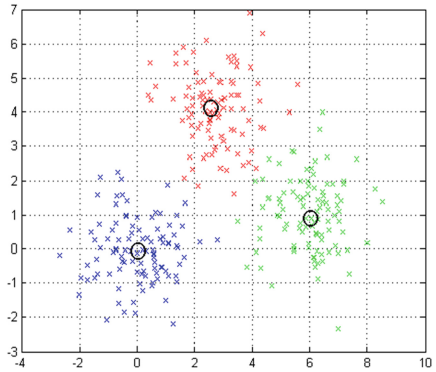


Fig. 5. Cluster plot for Dataset1 for ssFPA/DE

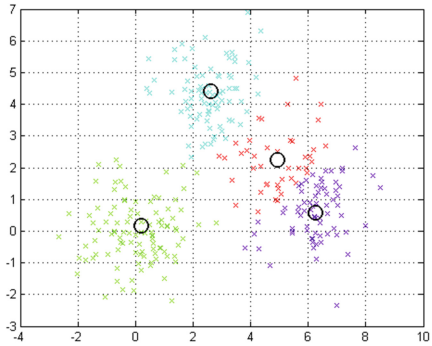


Fig. 6. Cluster plot for Dataset2 using DE

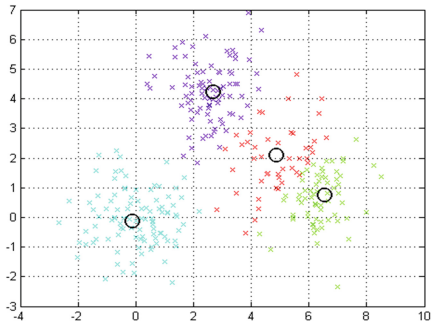


Fig. 7. Cluster plot for Dataset2 for 4 clusters using ssFPA/DE

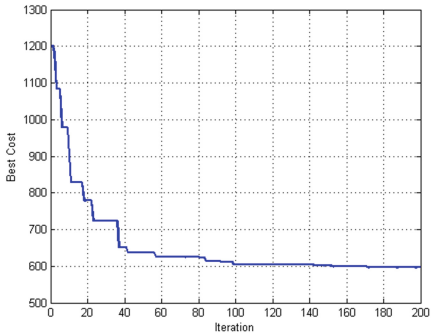


Fig. 8. Curve graph for Dataset1 for DE

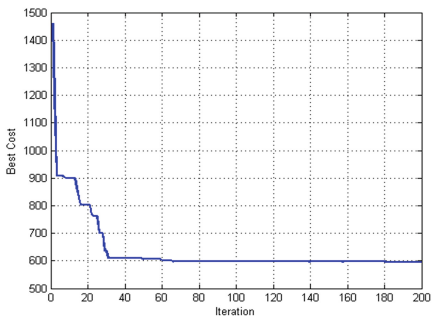


Fig. 9. Curve graph for Dataset1 using ssFPA/DE

8 Conclusion

The proposed technique is at an experimental stage. The hybrid algorithm was applied to different standard function and the results were compared. Work on clustering has been done only for the basic DE/best/1 strategy. Work on clustering is yet to be extended for the different standard functions of Ackeley, Bird, extended cube etc. and for different mutation strategies. After the results are obtained, a detailed comparative study shall be done to propose an efficient method.

References

1. Alves, V.S., Campello, R.J., Hruschka, E.R.: Towards a fast evolutionary algorithm for clustering. In: IEEE Congress on Evolutionary Computation, 2006, CEC 2006, pp. 1776–1783. IEEE (2006)
2. Storn, R., Price, K.: Differential evolution – a simple and efficient heuristic for global optimization over continuous spaces. J. Global Optim. **11**(4), 341–359 (1997)
3. Voges, K.E., Pope, N.K.L.: Rough clustering using an evolutionary algorithm. In: 2012 45th Hawaii International Conference on System Science (HICSS), pp. 1138–1145. IEEE (2012)
4. Jian-Xiang, W., Huai, L., Yue-Hong, S., Xin-Ning, S.: Application of genetic algorithm in document clustering. In: International Conference on Information Technology and Computer Science 2009, ITCS 2009, vol. 1, pp. 145–148. IEEE (2009)
5. Zhang, Z., Cheng, H., Zhang, S., Chen, W., Fang , Q.: Clustering aggregation based on genetic algorithm for documents clustering. In: IEEE Congress on Evolutionary Computation, CEC 2008, (IEEE World Congress on Computational Intelligence), pp. 3156–3161. IEEE (2008)
6. Zheng, Z., Gong, M., Ma, J., Jiao, L., Wu, Q.: Unsupervised evolutionary clustering algorithm for mixed type data. In: 2010 IEEE Congress on Evolutionary Computation (CEC), pp. 1–8. IEEE (2010)
7. Wang, R., Zhou, Y.: Flower pollination algorithm with dimension by dimension improvement. In: Mathematical Problems in Engineering, pp. 1–9 (2014)
8. Martínez-Estudillo, A.C., Hervás-Martínez, C., Martínez-Estudillo, F.J., García-Pedrajas, N.: Hybridization of evolutionary algorithms and local search by means of a clustering method. IEEE Trans. Syst. Man Cybern. Part B (Cybernetics) **36**(3), 534–545 (2005)
9. Zhou, Y., Wang, R., Luo, Q.: Elite opposition-based flower pollination algorithm. Neurocomputing, **188**, 294–310 (2016)

Distributed Clustering Scheme to Relieving Broadcast Storms in Vehicular Ad-Hoc Network

Abdelali Touil(✉) and Fattehallah Ghadi

Laboratory of Engineering Sciences (LabSI), Department of Computer Science, Faculty of Science, Ibn Zohr University, B.P 8106, Agadir, Morocco
abdelali.touil@edu.uiz.ac.ma, f.ghadi@uiz.ac.ma

Abstract. Nowadays, Intelligent Transportation Systems (ITS) aim to reduce accidents, fuel consumption and roads congestion, for this reason Vehicular Ad-hoc Networks (VANETs) technology as special case of ITS represent the future of the smart vehicles, the use of this one in order to improve road traffic safety will be great challenge. As can be seen, the number of cars worldwide is growing day after day. The deployment of VANETs technology which is composed of several types of communications has interested many researchers. So that to stabilize and handle this number of vehicles, we have focused our search on Quality of Service (QoS) and increased packet delivery ratio between vehicles by implementing Clustering scheme.

This paper describes a new distributed algorithm connectivity-based to form stable clusters. To evaluate our proposal scheme, we have conducted various simulations in OMNeT++ and SUMO simulators. The obtained results are promising in terms of packet delivery ratio and overhead.

Keywords: VANETs · Dynamic clustering · Connectivity · Stability · CH election · QoS

1 Introduction

For many years, the transportation systems in the world become increasingly a major issue, resulting in numerous accidents, deaths by injuries and traffic jam. Indeed, there are more and more vehicles, but with the same space to share and little budget for enhancing the infrastructure. The use of technology to make road traffic more safe and efficient is necessary. Fortunately, in recent years, many technologies are invented to collect information using sensors in a variety of forms: text, audio and video. The mass of collected data offer promising way to provide solutions to problems.

Vehicular Ad-hoc networks are the new emerging technology which offers many implementations on roads, namely: traffic optimization and monitoring, eco-friendly driving, cooperative driving, vehicle diagnostics and collision avoidance [1]. VANETs are basically a special application of Mobile Ad-hoc Network (MANETs) where vehicles communicate with each other the information about roads traffic.

However, due to the number of vehicles, velocity constraint, driver behavior, and frequently change of topology [2], VANETs exhibit characteristics that are very

A. Abraham et al. (eds.), *Proceedings of the 16th International Conference on Hybrid Intelligent Systems (HIS 2016)*, Advances in Intelligent Systems and Computing 552,
DOI 10.1007/978-3-319-52941-7_54

different from MANETs, and which causes frequent disconnection of communications and data loss.

So as to improve vehicular communications, the clustering is a technique of dividing a group of vehicles into a set of virtual groups called clusters; each cluster is managed by particular vehicle called cluster head; this one is selected according to specific criteria, which are generally the key elements to form stable clusters. Clustering improves the stability and quality of service (QoS) [3]. Depending on the characteristics of VANETs, the classic clustering schemes are not suitable for vehicular communications. Therefore, new clustering schemes should be invented particularly for VANETs [4].

In the current paper, we present a novel clustering connectivity-based scheme, which is adapted to form clusters and maintain their structure after each change in network topology. In order to implement this solution in VANETs, it should be distributed, with no any central infrastructure.

The paper is organized as follows. Section 2 introduces Vehicular Ad-hoc Networks technology. Section 3 presents our solution and different parties: Clusters formation and maintenance, while in Sect. 4 we evaluate the efficiency of our solution in comparison with one existing algorithms. Finally, Sect. 5 concludes the paper.

2 VANETs Technology

2.1 DSRC

Dedicated short range communication is a wireless technology, generally invented for establish the communication between vehicles. The Federal committee for communications granted in USA 75 MHz of spectrum at 5.9 GHz for the DSRC [10]. Also, the European Telecommunication Standards Institute assigned in Europe 30 MHz of spectrum at 5.9 GHz for transmission DSRC data.

The structure of the DSRC is constituted of two different channels: Control channel CCH reserved to send and the receipt safety messages and the service channel used for data messages (Figs. 1 and 2).

172 5.860 GHz	SCH 174 5.870 GHz	SCH 176 5.880 GHz	**CCH** 178 5.890 GHz	SCH 180 5.900 GHz	SCH 182 5.910 GHz	184 5.920 GHz

Fig. 1. DSRC in USA

SCH 172 5.860 GHz	SCH 174 5.870 GHz	SCH 176 5.880 GHz	SCH 178 5.890 GHz	**CCH** 180 5.900 GHz

Fig. 2. DSRC in Europe

2.2 WAVE

Wireless Access in Vehicular Environment (WAVE) technology also known as 802.11p [5, 6] is a protocol developed for vehicle-to-vehicle (V2V) and vehicle-to-Infrastructure (V2I) communications. In order to share safety and non-safety data, WAVE is based on Orthogonal Frequency Division Multiplexing (OFDM) method. Basically it uses a bandwidth of 75 MHz at 5.9 GHz. This last is divided into seven small channels of 10 MHz each. The channels 172, 174, 176, 180, 184 and 186 are reserved for safety and non-safety data. However, channel 178 is reserved for safety data.

So as to establish communication between vehicles themselves without any support of infrastructure and vehicles and infrastructure, each vehicle should be equipped with two units, OnBoard Unit (OBU) and Application Unit (AU). OBU is a wireless device based on IEEE 802.11p radio frequency channel, usually installed in each vehicle. Its main role is to exchange safety and non-safety information with other OBUs or with RSUs in its neighborhoods. It may additionally include another extension to connect with other technologies [7]. Although, AU is an intermediary device between the driver and the OBU, it runs applications in order to display the information to drivers.

However, RoadSide Unit (RSU) is wireless device installed on the road. It is based on IEEE 802.11p technology. RSU exchange information with vehicles through OBUs or with other RSUs. RoadSide Unit also provides other services for vehicles, namely: Internet, Car-to-home communication, calling emergency service, etc. [8].

The main problem that stalled the performance of VANETs applications is when vehicles communicate with each other on congested points mostly on urban roads. They broadcast periodically information about their velocity, location and neighbors table, which create a broadcast storm of messages resulting in network saturation. In some cases, this failure causes a situation of denial of service (DoS), overhead and decreases the packet delivery ratio (Fig. 3).

For this purpose, the primary objective of this paper is to provide a solution to this problem. Through the use of clustering scheme in the network, only cluster heads will be able to broadcast information.

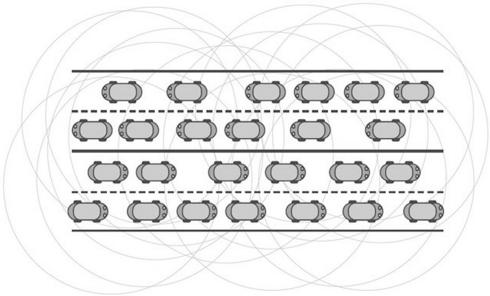


Fig. 3. Saturation of network by broadcast storm [9]

3 Proposed Clustering Scheme

In order to implement clustering in VANETs, the first issue is how to form clusters using clustering algorithm. In terms of sharing tasks between vehicles, four categories of vehicles have been defined: Undefined, Cluster head, Cluster member and Gateway (Fig. 4).

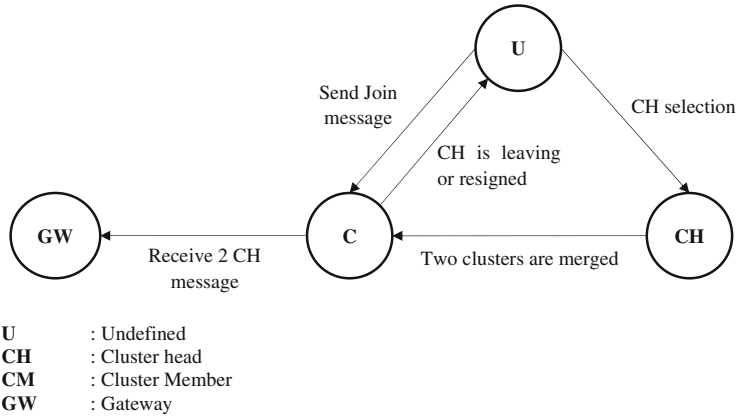


Fig. 4. Proposed clustering scheme

3.1 Cluster Head Selection

To form clusters, each vehicle calculates its degree to determine the eligibility as a cluster head. The metric of vehicle is defined to increase the stability of the cluster structure and Quality of Service. As a result, an elected cluster head is expected to stay connected with its members for the longest period of time. Therefore, vehicles having high degree difference value are more qualified for being cluster head.

The degree of a given vehicle is the number of the neighbor's vehicles connected to this one $|N(v)|$. It is determined and is represented as $deg(v)$, and the degree difference is calculated as follow:

$$\Delta_v = |\deg(v) - \delta| \quad (1)$$

Where δ is the number of vehicle that a cluster head can serve, and Δ_v is the degree difference.

3.2 Clusters Formation and Maintenance

An efficient clustering algorithm should provide a stable clusters topology with minimal communications overhead and complexity. The efficiency our algorithm depends on the number of clusters formed and the quantity of packet Delivery Ratio. That's way the presented clustering scheme regroups the vehicles by high connectivity, as a main

factor of stability. This last must be deployed and executed asynchronously by each vehicle in the network to achieve minimal clustering overhead with more Quality of Service.

- Cluster Formation

Each vehicle *v* added to the network or changed the cluster executes the procedure of initialization in order to determine its own role, namely: Cluster head, Member or Gateway. If there is at least a cluster head with high connectivity and in the same direction in its neighborhood table, then v will join it. Otherwise it will be a cluster head.

- Cluster Maintenance

When a vehicle v detect that a vehicle w located in its neighborhood table exceeds the time fixed for it (TTL) without receive any beacon message, u will be automatically deleted from the neighborhood table, and it will be deleted also in the cluster table if v is a cluster head. If vehicle v is a member and u is its cluster head, v checks if there is at least one cluster head z with a lower weight in the same direction to join it. Otherwise, it becomes a cluster head.

4 Simulation Results

To evaluate the performance of our solution we will compare with Packet Delivery Ratio and the average number of generated clusters. We have choose three simulators: OMNet++ simulator [11] and SUMO [12] tools. So as to join these two tools, we resorted to the use Veins frame work [13].

Figure 5 presents the impact of speed of vehicles on packet delivery ratio. The figure shows that the changes in speed between 60 to 100 km/s, the packet delivery ratio will slightly change.

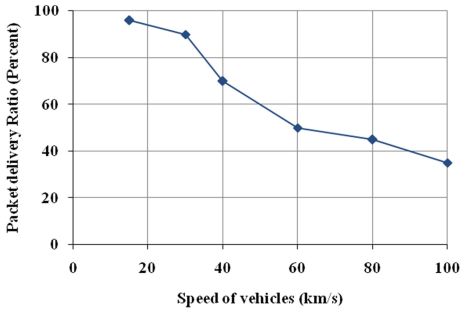


Fig. 5. Packet Delivery Ratio in proposed scheme with considering the speed of vehicles

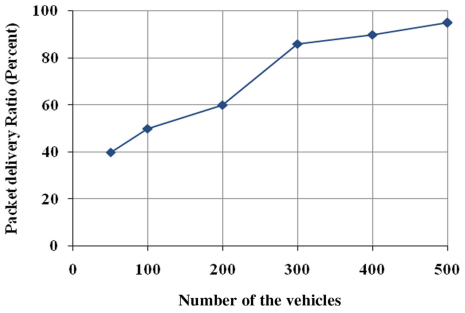


Fig. 6. Packet Delivery Ratio in proposed scheme with considering the number of vehicles

Figure 6 presents the impact of the number of vehicles on the packet delivery ratio. As the figure shows the increase in the number of vehicles causes in parallel the increase of packet delivery ratio.

In this simulation, we made a comparison between an already existing algorithm solution in building clusters which is Distributed Mobility Algorithm Clustering [14]. DMAC is a cluster construction algorithm one-hop, cluster-heads are selected by the higher or lower Id. Generally DMAC considers that each vehicle has a unique ID that differentiate it from other vehicles and does not considered the direction of vehicles. The identifier is integer for facilitate the comparison, in this simulation we'll deem that the cluster head is the vehicle that has the lower ID.

As presented in Fig. 7, the combined metrics generates fewer clusters and the number of clusters in the network is more stable, by against the number of clusters generated by DMAC remains in the increase.

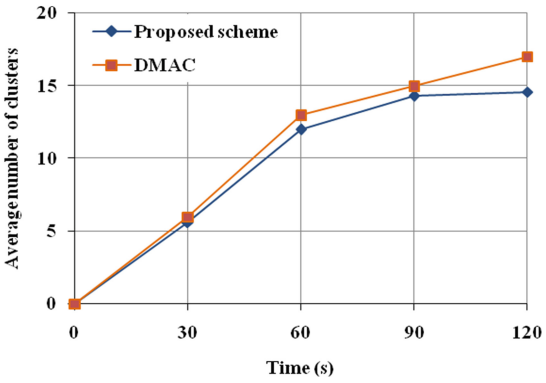


Fig. 7. Number of clusters, proposed scheme compared to DMAC

5 Conclusion

In this paper, we have presented in Sect. 3 the advantages of clustering scheme to enhance the stability and QoS of the network. For example, grouping vehicles into clusters make a network topology easy to maintain, and grouping vehicles of the same direction increase the stability of the network. In this paper, we consider a transmission of data between the vehicles. Vehicles nearby are grouped into a cluster in which some vehicles are selected as cluster heads to manage the clusters. These cluster heads serve as a router that enables other vehicles to communicate with each other.

References

1. Raw, R.S., Kumar, M., Singh, N.: Security challenges, issues and their solutions for VANET. Intl. J. Netw. Secur. Its Appl. **5**(5), 95 (2013)
2. Al-Sultan, S., Al-Doori, M.M., Al-Bayatti, A.H., Zedan, H.: A comprehensive survey on vehicular Ad Hoc network. J. Netw. Comput. Appl. **37**, 380–392 (2014)
3. Bali, R.S., Kumar, N., Rodrigues, J.J.: Clustering in vehicular ad hoc networks: taxonomy, challenges and solutions. Vehicular Commun. **1**(3), 134–152 (2014)
4. Vodopivec, S., Bester, J., Kos, A.: A survey on clustering algorithms for vehicular ad-hoc networks. In: 35th International. Conference on Telecommunications and Signal Processing (TSP), pp. 52–56 (2012)
5. Chen, Q., Jiang, D., Delgrossi, L.: IEEE 1609.4 DSRC multi-channel operations and its implications on vehicle safety communications. In: 2009 IEEE Vehicular Networking Conference (VNC), pp. 1–8. IEEE, October 2009
6. Jiang, D., Delgrossi, L.: IEEE 802.11 p: towards an international standard for wireless access in vehicular environments. In: Vehicular Technology Conference 2008, VTC Spring 2008, pp. 2036–2040. IEEE, May 2008
7. Xiang, W., Gozalvez, J., Niu, Z., Altintas, O., Ekici, E.: Wireless access in vehicular environments. EURASIP J. Wireless Commun. Netw. **2009**(1), 1–2 (2009)
8. Al-Sultan, S., Al-Doori, M.M., Al-Bayatti, A.H., Zedan, H.: A comprehensive survey on vehicular Ad Hoc network. J. Netw. Comput. Appl. **37**, 380–392 (2014)
9. Sahasrabudhe, M.S., Chawla, D.M.: Survey of applications based on vehicular ad-hoc network (VANET) framework. IJCSIT Intl. J. Comput. Sci. Inf. Technol. **5**(3), 3937–3942 (2014)
10. Regan, A.: The ACCESS almanac: vehicular ad hoc networks: storms on the horizon. ACCESS Magazine (2013)
11. Rawashdeh, Z.Y., Mahmud, S.M.: A novel algorithm to form stable clusters in vehicular ad hoc networks on highways. EURASIP J. Wireless Commun. Netw. **2012**(1), 15 (2012)
12. Ahmed, S.A., Ariffin, S.H., Fisal, N.: Overview of wireless access in vehicular environment (WAVE) protocols and standards. Indian J. Sci. Technol. **6**(7), 4994–5001 (2013)
13. Zeadally, S., Hunt, R., Chen, Y.S., Irwin, A., Hassan, A.: Vehicular ad hoc networks (VANETS): status, results, and challenges. Telecommun. Syst. **50**(4), 217–241 (2012)
14. Basagni, S.: Distributed clustering for ad hoc networks. In: Proceedings of the Fourth International Symposium on Parallel Architectures, Algorithms, and Networks (I-SPAN 1999), pp. 310–315 (1999)

PSO for Job-Shop Scheduling with Multiple Operating Sequences Problem - JS

Sana Khalfa[1(✉)], Nizar Rokbani[2,4], Achraf Jabeur Telmoudi[1,4], Imed Kacem[3], Lotfi Nabli[1,5], and Alaoui Mdaghri Zoubida[6]

[1] LARATSI-Lab: ENIM, University of Monastir, Monastir, Tunisia
sana.khalfa.tn@ieee.org, achraf_telmoudi@yahoo.fr, Lotfinabli@yahoo.fr
[2] REGIM-Lab: ENIS, University of Sfax, Sfax, Tunisia
nizar.rokbani@ieee.org
[3] LCOMS-Lab, Université de Lorraine, Nancy, France
[4] Higher Institute of Applied Sciences and Technology of Sousse, University of Sousse, Sousse, Tunisia
[5] National School of Engineering of Monastir, University of Monastir, Monastir, Tunisia
[6] Faculty of Science Rabat, University Mohammed V, Rabat, Morocco
zoubidaalaouimdaghri@gmail.com

Abstract. This paper focus on a complex problem of job shop scheduling where each jobs have a multiple possible operations sequences. The resolution of this type of problem has not been treated in the literature. To solve this, a new algorithm based on Particle Swarm Optimization Global Velocity (PSOVG) was proposed. The objectif is to minimize the makespan. The simulation results show the efficiency of our proposed approach.

Keywords: Job-shop scheduling · Multiple operations sequences · PSOVG · Makespan

1 Introduction

Job-Shop scheduling problem (JSSP) is among the hardest combinatorial optimization problems. During the last three decades, the problem has captured the interest of a significant number of researchers and a lot of literature has been published, but no efficient solution algorithm has been found yet for solving it to optimality in polynomial time [1]. Scheduling is the allocation of shared resources over time to competing activities. It has been the subject of a significant amount of literature in the operations research field. Emphasis has been on investigating machine scheduling problems where jobs represent activities and machines represent resources. This problems are complex, NP-Hard and impossible to solve them with classical methods [2].

Heuristic and metaheuristic approaches are developed by researchers to solve this type of problem [3]. A classification of the scheduling techniques is given in Jain (1998), MacCathy and Liu, 1993; Esswein, 2003. Jonhson [4] developed a heuristic to solve two and three stage production scheduling problem. Fisher (1986) exploited the Lagrangian

A. Abraham et al. (eds.), *Proceedings of the 16th International Conference on Hybrid Intelligent Systems (HIS 2016)*, Advances in Intelligent Systems and Computing 552,
DOI 10.1007/978-3-319-52941-7_55

relaxation technique to obtain a more efficient method. Many researchers applied branch and bound algorithm to solve the JSSP [5]. A more sophisticated method called shifting bottleneck (SB) has been shown to be very successful [6]. Additionally, stochastic approaches such as simulated annealing (SA), tabu search [7, 8] and genetic algorithms (GAs) have been recently applied with good success [9].

Zhigang Lian et al. [10] presented a similar PSO algorithm (SPSOA) for job-shop scheduling to minimize the makespan. In these perspectives, we propose an intelligent algorithm for the JSSP based on the PSOGV. PSOVG is an evolutionary computation techniques based on swarm intelligence. This method can be classified as stochastics methods. The goal is to improve the existing situation by moving part according to predefined rules, to achieve the desired overall solution [11].

The remainder of the paper is organized as follows. Section 2 will describe the problem and define the notation. Section 3 presents the PSOVG algorithm applied to JSSP with multiple operating sequences. In Sect. 4, simulation and results are discussed. Finally, conclusion and perspectives are made in Sect. 5.

2 Job-Shop Scheduling Problem with Multiple Operating Sequences, JSSP-MOS

Our research work is to develop an approach that minimizes the makespan of a problem of job-shop scheduling with multiple operating sequences. Given its complexity, this problem can be considered as NP-hardest problem type. The objective of our works is to minimize the maximum completion time of the final operation in the schedule of $n \times m$ operations. In our work, we focus on the problem witch workshop resources are machines and tasks are to schedule operations to be performed on these machines. In our case tasks have multiple operating sequences. There are several methods of resolutions: exact methods, heuristics, meta-heuristics. In this paper, we will use Particle Swarm Optimization (PSO) algorithm.

2.1 JSSP-MOS Formulation

Considering job-shop scheduling with n jobs where each job contains m operating ranges and each operating range through a number k machines in a specific order. The number of lines and machines for each range is fixed as well as the sequence of operations of each job on each machine. The tasks are non-preemptive, they are executed only once and can not interrupted before they are fully completed. The thing that makes our problem more complicated is that every job i have multiple operating sequences possible for product realization. The number of operating sequences is not constant data.

In our approach, we use the PSOVG to generate all a random solution to make evolving to reach a live of optimatile. The optimal solution is the schedule, which has the minimum of Makespan called *MinCmax*. To approve our approach, we compared the results find with random results. Simulations show the effectiveness of our algorithm.

The problem may be formulated as follows. There is a set of n jobs that process on m machines. The set machines Mj = {M1, M2,…, Mm} are determinate on advance. Each job Ji consists of Gi,k operating sequences. Each Gi,k operating sequences have nj tasks. Number of tasks for each operating sequences are given depending on the case.

2.2 JSSP-MOS Hypotheses and Notations

In this section, we will describe the different variables used in our work:

1. NumOpSJob [1, i] is a matrix formed by one row and i colomn where each colomn contains the number of operating sequences (OpS) of each job j
2. J_i: Job number i
3. $G_{i,k}$: Operating Sequence number k of a job i
4. M_j: Machine number j

Hypotheses considered in this paper are: Jobs are independent from each other Machines are independent from each other One machine can only execute one operation.

There are the precedence constraints among operations of each job.

3 PSO for Job Shop Scheduling

Particle Swarm Optimization is considered as a stochastic heuristic optimization technique. This technique is used to optimize complex problems that known as difficult to solve [12].

3.1 The PSO Algorithm

PSO is an optimization method developed by Kennedy and Eberhart and inspired by bird flocks' behavior when they fly to look for their foods [13]. The groups of individuals, called particles, have the same dimension corresponding to the number of problem parameters. Generally, particles are randomly dispersed in the search space. Then, the fitness value is calculated by each particle according to the objective function of the problem.

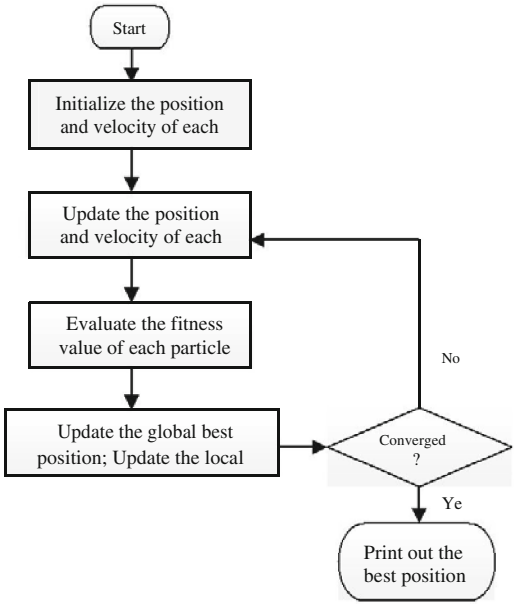


Fig. 1. The flowchart of PSO algorithm.

In PSO, each particle i has a position Xi and velocity V*i*. Equations (1) and (2) describe these terms.

$$\overrightarrow{V_{i+1}} = w * \overrightarrow{V_i} + C1 * rand() * (\overrightarrow{X_l} - \vec{X}) + c2 * rand(\overrightarrow{X_y} - \overrightarrow{X_i}) \quad (1)$$

$$\overrightarrow{X_{i+1}} = \overrightarrow{X_i} + \overrightarrow{V_{i+1}} \quad (2)$$

Where C1 is the cognitive moderation parameter and C2 is the social moderation parameter. C1 allows to take into account the effort of partical local best while C2 modirates the motion of the partical toward the global best.

To solve a problem the classical PSO processing consists in repeating the following steps until a stop condition is achieved. Figure 1 presents the flowchart of the PSO algorithm [14].

3.2 The Proposed PSO-VG Algorithm

To solve this problem an algorithm based on PSO global velocity have been proposed. Figure 2 presents the flowchart of this algorithm.

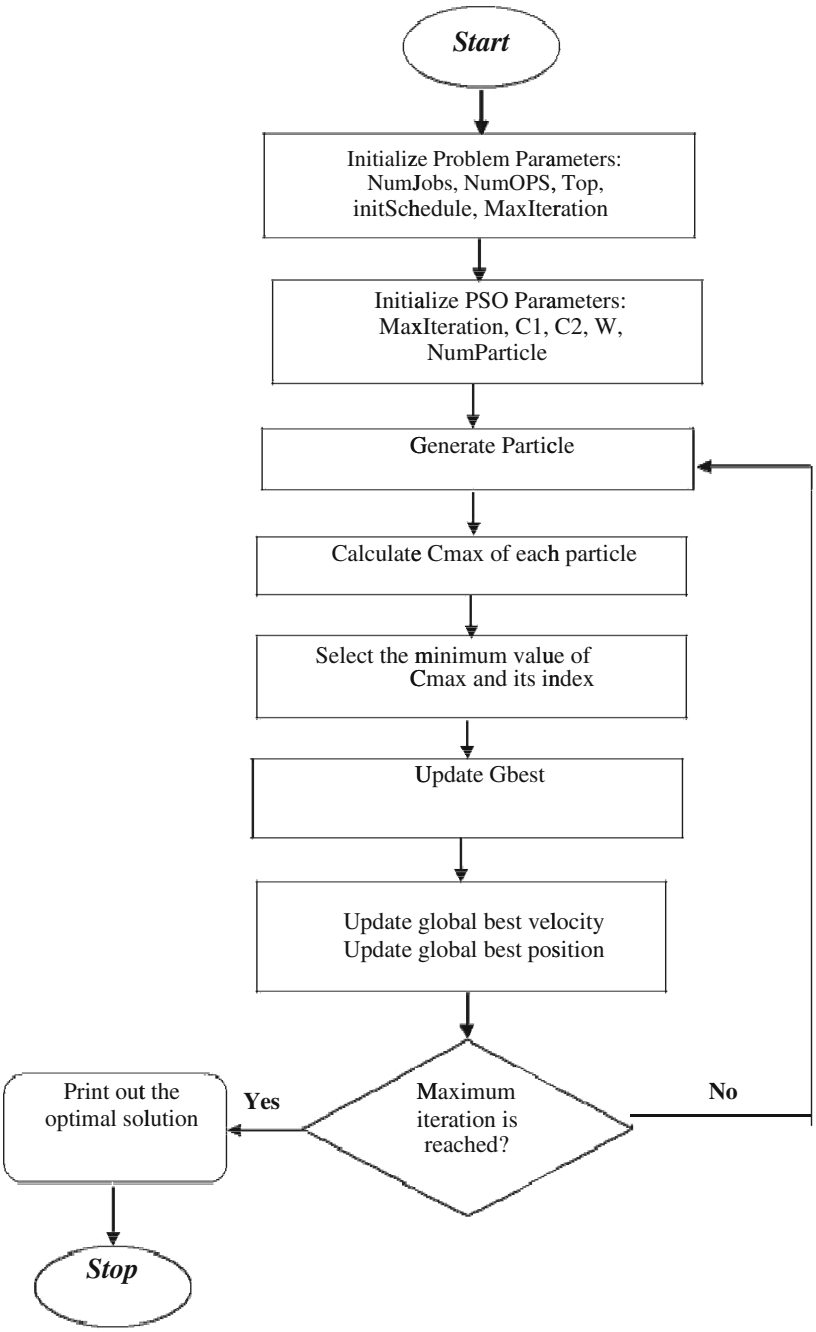


Fig. 2. The flowchart of PSOVG algorithm.

4 Simulation and Results

Simulation results are serve on the following problems.

In our example, we will fixed the number of jobs and machines and varing the operating sequences number: 3 jobs (J1 J2 and J3) and 3 machines (M1 M2 et M3) with numbers of operating sequences for each jobs (NumOpSJob) are defined on advanced us this cases follow.

4.1 Case1: NumOpSJob = [3 2 2]

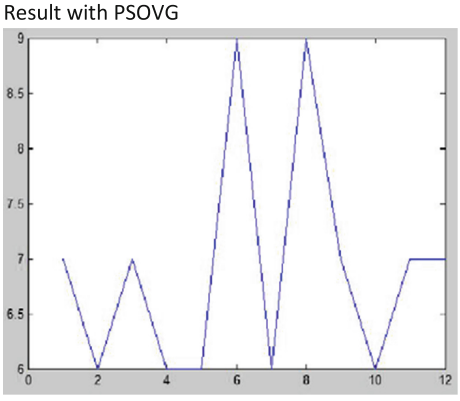
Jobs = [(1 2 3;3 2 0;1 2 0);(2 1 0;2 3 0);(1 3 0;2 3 0)]
Tops = [(1 2 3;3 2 0;1 2 0);(2 1 0;2 3 0);(1 3 0;2 3 0)]

(Tops: operating time for each tasks)

Figures 3 and 4 are composed by the x-axis which represents the Cmax Value and the y-axis represents the number of iteration. The optimal value of Cmax with PSOVG is found at the 2nd iteration then the optimal value of Cmax with Random is found after 9 iterations.

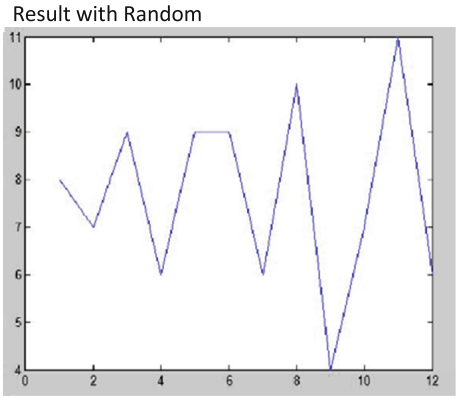
We can conclude that the optimal value of Cmax using PSOVG which is near to the best solution (OptimalCmax = 4) is reached with better execution time than Cmax using Random. Therefore, Cmax with PSOVG converge rapidly to the optimum.

Knowing that PSO is used for complex problem with large size, so we have applied our algorithm using a case with large number of operating sequences. Results in Table 1 demonstrate the effectiveness of our approach (Figs. 5 and 6).



OptimalCmax=6
OptimalOperatingSequencesChoice=[2 1 1]
OptimalScheduling=[(3 2 0) ;(2 1 0) ;(1 3 0)]

Fig. 3. Historic Cmax with PSOVG.



OptimalCmax=4
OptimalOperatingSequencesChoice=[3 1 1]
OptimalScheduling=[(3 2 0) ;(2 1 0) ;(1 3 0)]

Fig. 4. Historic Cmax with Random.

4.2 Case2: NumOpSJob = [24 16 16]

Table 1. The results obtained by the PSOVG-MOS and compared with random results.

Result with PSOVG				Result with random		
Test num	Opt Cmax	Choice of OpSeq	Schedule sequences	Opt Cmax	Choice of OpSeq	Schedule sequences
1	6	[10 7 8]	[(1 2 0);(1 2 0);(1 3 0)]	4	[11 7 15]	[(2 1 0);(1 2 0);(1 3 0)]
2	9	[5 4 5]	[(2 3 0);(2 3 0);(1 2 0)]	4	[24 10 6]	[(1 2 0);(1 3 0);(2 1 0)]
3	5	[6 8 5]	[(1 3 0);(2 1 0);(1 2 0)]	4	[10 15 8]	[(1 2 0);(2 1 0);(1 3 0)]
4	5	[11 10 6]	[(2 1 0);(1 3 0);(2 1 0)]	4	[18 14 1]	[(2 1 0);(1 2 0);(1 3 0)]
5	7	[8 8 4]	[(1 2 3);(2 1 0);(3 2 0)]	4	[3 3 6]	[(1 2 0);(1 3 0);(2 1 0)]
6	10	[8 6 5]	[(1 2 3);(3 2 0);(1 2 0)]	4	[11 7 15]	[(2 1 0);(1 2 0);(1 3 0)]
7	8	[5 7 3]	[(2 3 0);(1 2 0);(1 2 3)]	4	[3 3 6]	[(1 2 0);(1 3 0);(2 1 0)]
8	4	[4 7 8]	[(2 1 0);(1 2 0);(1 3 0)]	4	[17 1 8]	[(1 2 0);(2 1 0);(1 3 0)]
9	4	[10 10 6]	[(1 2 0);(1 3 0);(2 1 0)]	4	[11 14 8]	[(2 1 0);(1 2 0);(1 3 0)]
10	4	[10 8 8]	[(1 2 0);(2 1 0);(1 3 0)]	4	[17 15 8]	[(1 2 0);(2 1 0);(1 3 0)]

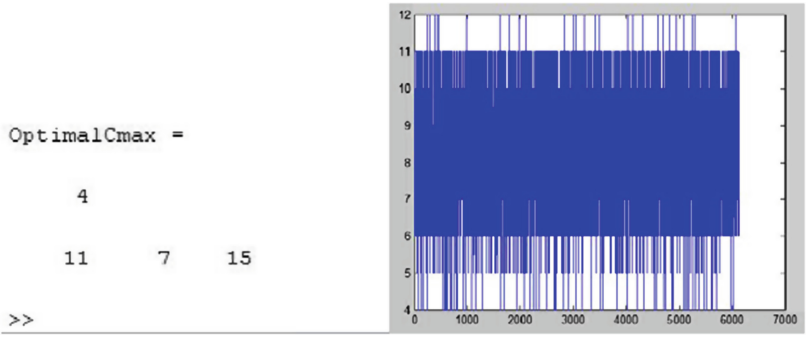


Fig. 5. Value of Cmax with Random.

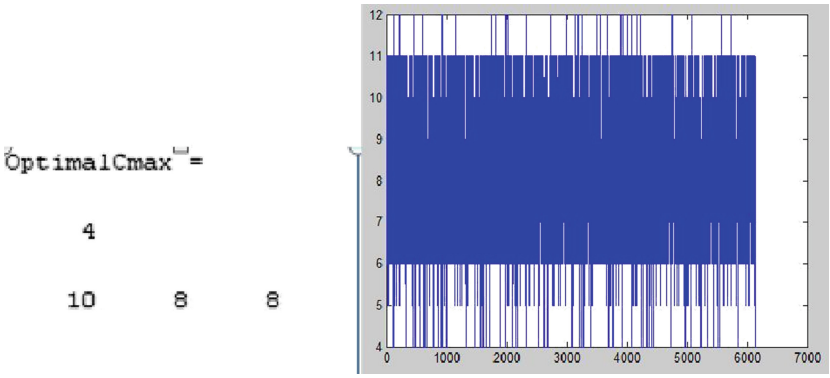


Fig. 6. Value of Cmax with PSOVG.

5 Conclusion and Perspectives

In this paper, an algorithm based on the PSO-VG has involved to solve the Job shop scheduling problem with multiple sequences. The main contribution consisted to find the optimal schedule either the number of possible range for each job; this under precedence and non preemptions constraints. The proposed methods shown it capacity in founding the minimum Makespan for simple test bench within a limited iteration number. Meaning that the proposed method has a very limited processing time. More investigations are needed to confirm convergence for large set JSS problems. The impact of PSO VG on convergence need also to be investigated as well as the existence of a good set of parameters of PSO and especially PSO-VG for this class of problems.

Acknowledgments. The authors would like to acknowledge the financial support of this work by grants from General Direction of Scientific Research (DGRST), Tunisia, under the ARUB program.

References

1. Cheng, R., Gen, M., Tsujimura, Y.: A tutorial survey of job-shop scheduling problems using genetic algorithms—I representation. Comput. Ind. Eng. **30**(4), 983–997 (1996)
2. Liu, T.K., Chen, Y.P., Chou, J.H.: Solving distributed and flexible job-shop scheduling problems for a real-world fastener manufacturer. Access IEEE **2**, 1598–1606 (2014)
3. Fahmy, S., Balakrishnan, S., ElMekkawy, T.: Deadlock prevention and performance oriented supervision in flexible manufacturing cells: a hierarchical approach. Robot. Comput.-Integr. Manuf. **27**(3), 591–603 (2011)
4. Johnson, S.M.: Optimal two and three-stage production schedules with setup times included. Naval Res. Logistics Q. **1**(1), 61–68 (1954)
5. Brucker, P., Jurisch, B., Sievers, B.: A branch and bound algorithm for the job-shop scheduling problem. Discrete Appl. Math. **49**, 107–127 (1994)

6. Adams, J., Balas, E., Zawack, D.: The shifting bottleneck procedure for job shop scheduling. Manage. Sci. **34**(3), 391–401 (1988)
7. Dell'Amico, M., Trubian, M.: Applying tabu search to the job-shop scheduling problem. Ann. Oper. Res. **41**, 231–252 (1993)
8. van Laarhoven, P.J.M., Aarts, E.H.L., Lenstra, J.K.: Job shop scheduling by simulated annealing. Oper. Res. **40**(1), 113–125 (1992)
9. Bierwirth, C.: A generalized permutation approach to job shop scheduling with genetic algorithms. Oper. Res. Spektrum **17**, 87–92 (1995)
10. Lian, Z., Jiao, B., Gu, X.: A similar particle swarm optimization algorithm for job-shop scheduling to minimize makespan, pp. 1008–1017 (2006)
11. Shi, Y., Eberhart, R.: Empirical study of particle swarm optimization. In: Proceedings of congress on evolutionary computation, pp. 1945–1950 (1999)
12. Bonyadi, M.R., Michalewicz, Z.: Particle swarm optimization for single objective continuous space problems: a review. Evol. Comput. (2016, in press)
13. Kennedy, J., Eberhart, R: Particle swarm optimization. In: IEEE International Conference on Neural Networks, pp. 1942–1948 (1995)
14. Kefi, S., Rokbani, N., Krömer, P., Alimi, A.M.: A new ant supervised-PSO variant applied to traveling salesman problem. In: The 15th International Conference on Hybrid Intelligent Systems (HIS), Seoul, 16–18 November, South Korea, pp. 87–101 (2015)

Impact of Ant Size on Ant Supervised by PSO, AS-PSO, Performances

Sonia Kefi[1(✉)], Nizar Rokbani[1,2], and Adel M. Alimi[1]

[1] REGIM-Lab: Research Groups in Intelligent Machines, University of Sfax, ENIS, Sfax, Tunisia
{sonia.kefi, nizar.rokbani, adel.alimi}@ieee.org
[2] High Institute of Applied Sciences and Technology of Sousse, University of Sousse, Sousse, Tunisia

Abstract. AS-PSO, ANT Supervised by PSO is hybrid hierarchical meta-heuristic optimization method where PSO optimizes ANT parameters to enhance its performances. In this paper, a focus is made on the impact of the ACO swarm size on AS-PSO performances for the Traveling Salesmen Problem (TSP) where AS-PSO is already known as a relevant solver. Investigations used the AS-PSO-2Opt with both inertia weight AS-PSO and Standard AS-PSO. To demonstrate the effects of ant numbers on AS-PSO-2Opt method, a selected set of test benches form TSPLIB, berlin52, st70 and eli101 was used. In this experimental study of the ant number is waved from five to the city number of each selected test benches. Therefore, experimental results showed that the best swarm size is equal to 20 and gives the best solution for all test benches.

Keywords: Metaheuristic optimization · ACO · PSO · AS-PSO-2Opt · TSP · Swarm size

1 Introduction

The Travelling Salesman Problem (TSP) consists to find a shortest path for a salesman to visit n cities. Generally, the main objective of this problem is to minimize the total distance and total tour time. The TSP problem is a discrete combinatorial optimization problem and NP-complete problem so the complexity is not polynomial [1].

Several methods are used to solve TSP including *exact methods* and *metaheuristic methods*. From literature, *metaheuristic methods* [2] give good solution. Several metaheuristics methods are developed and based on solution of the problem as neighborhood search methods like Taboo Search [3], Genetic Algorithm [4], Particle Swarm Optimization [5] and Ant Colony optimization [6], etc.

In general, improvement heuristics produce better results than the construction heuristics. The general approach of hybridization is to use a construction heuristic to generate a solution and then apply a heuristic to improve this solution and optimize it.

For this reason, this paper presents a hybrid metaheuristic which was proposed, called Ant Supervised by particle swarm optimization (AS-PSO). This method is based on ACO and PSO algorithms to solve routing problem especially TSP. In [7, 8], the authors have developed several variants of Ant supervised by PSO (AS-PSO) like the

A. Abraham et al. (eds.), *Proceedings of the 16th International Conference on Hybrid Intelligent Systems (HIS 2016)*, Advances in Intelligent Systems and Computing 552,
DOI 10.1007/978-3-319-52941-7_56

simplified AS-PSO using the simplified variant of PSO. The main objective of these methods, the optimization of ACO parameters (α, β). Then, an extended work of AS-PSO was presented in [9]. In this variant, the authors try to optimize the four ACO parameters (α, β, τ, ρ). Another closed proposal is presented in [10].

In [11], the authors have proposed a hybrid variant based on local search algorithm called AS-PSO-2Opt method. This method allows optimizing the three ACO parameters (α, β, ρ) by PSO algorithm. Furthermore, the 2Opt algorithm is used to avoid the premature stagnation of AS-PSO and to decrease the possibility of filling in local minimum. From the obtained results showed that the AS-PSO-2Opt is better than other metaheuristic methods given using Genetic Algorithm [12] and Neural Network [13] for several test benches. In addition, AS-PSO-2Opt finds global best solutions than others variants of ACO, AS-PSO [7, 8] and Extended AS-PSO [9].

To improve the results of AS-PSO-2Opt for large test benches, a standard variant of AS-PSO-2Opt (SAS-PSO) [14] to solve routing problem as TSP is proposed. This variant is based on standard PSO. It main objective is to solve the problem of PSO. In fact, in PSO, particles can exceed the boundaries of the search space and they can fall in the local minimum. Therefore, standard PSO fixed a lower bound and an upper bound of variables for velocity and particle position. The simulation results of this method demonstrate the efficiency of SAS-PSO to solve the TSP compared to other methods from literature. This comparison is performed with several TSP benchmarks from TSPLIB [15].

This paper presents a new study of the impact of ACO swarm size on AS-PSO, which was not investigated in our previous work. To demonstrate the effects of ant numbers on the performance of AS-PSO-2Opt method, a several experiments are made on many TSP benchmarks. The best number of ants chosen give the optimal solutions.

This paper is organized as follows: In Sect. 2, we present an overview of PSO, ACO and AS-PSO-2Opt method. In Sect. 3, we describe the detailed study of ant number of ACO algorithm and we realize some experimentation. Finally, in Sect. 4, we summarize the results and we emphasize on the relevance of ACO.

2 Methods and Materials

In this section, we present algorithms that we have been used to describe the AS-PSO-2Opt approach.

2.1 Particle Swarm Optimization

PSO is an optimization algorithm taken inspirations from optimal behavior or animals groups. The social population is called the swarm and each individual, in the swarm, is known as particle [5].

At first, in the search space, particles are randomly dispersed; then, to select the best solutions, each particle calculate a fitness value using an objective function. The particle are supposed to move according to the Eqs. (1) and (2) until the end of processing conditions.

$$V_i\ (\mathrm{t}\ +\ 1) = w.V_i\ (\mathrm{t}) + c1.r1.(plbest_i\ (\mathrm{t}) - X_i(t)) + c2.r2.(pgbest(t) - X_i(t)) \quad (1)$$

$$X_i(t+1) = X_i(t) + V_{i+1}(t+1). \quad (2)$$

Where V_i is the velocity vector of i^{th} particle and X_i is i^{th} particle's location, in the next iteration. w is the inertia weight parameter which it moderates the particle current position. $c1$ adjusts the cognitive particle behavior, $c2$ controls the social particle behavior. $r1$ and $r2$ are random factors belonging to [0–1]. The $plbest_i$ and the $pgbest$ is the global best solution found.

2.2 Ant Colony Optimization

The ACO algorithm is an optimization method, it was proposed by Dorigo et al. [6]. This method was inspired from ant colony behaviors. The authors observed ants' behaviors in real nature and the possibility of each ant to achieve the shortest path among their nest and food source. The first variant of ACO is the Ant System algorithm [16], which is implemented, in our proposed method. The AS algorithm operates as iterative algorithm and its best solution is considered as the shortest route. The ant in city, i, selects and move to city, j. This selection is determined by the probability, P_{ij}, given in (3).

$$p_{ij}^{k}(t) = \begin{cases} \frac{[\tau_{ij}(t)]^{\alpha}.[\eta_{ij}]^{\beta}}{\sum_{k\in\Omega i}[\tau_{ik}(t)]^{\alpha}.[\eta_{ik}]^{\beta}} & \text{if j} \in \Omega i \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

Where τ_{ij} designates the amount of pheromone between i and j cities; Ωi stands for the cities for the k^{th} ant in iteration t and η_{ij} is computed according to Eq. (4):

$$\eta_{ij} = \frac{1}{dij}. \quad (4)$$

where dij is the distance between cities i and j.

2.3 The AS-PSO-2Opt Method

The initial step of AS-PSO-2Opt algorithm, ants are randomly dispersed in cities. Then, the amount of pheromones are affected to all arcs. All ants complete their first tour taking into account distances from the city. The ACO parameters α and β in (3) are determined and optimized using the two equations of PSO (1) and (2). The fitness function of PSO, it is calculate using the Euclidean distance of TSP that it gives the tour length. The result of our method is considered as the optimal tour of the TSP. When all the ants finish the construction of their tours, the pheromone trails are updated where the parameter ρ, pheromone evaporation rate, is determined by the two equations of PSO (1) and (2). After evaporation of pheromone, all ants, randomly dispersed in cities, deposit pheromone on the arcs, which they have crossed in their tour.

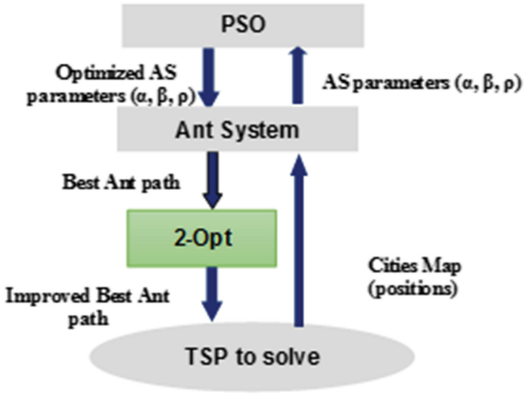


Fig. 1. AS-PSO-2Opt architectural overview.

After obtaining the best tour, an improvement local search algorithm has been used to improve the initial solution named 2-Opt algorithm. This improvement consists to remove two connections in the tour, to reconnect them with other means and then, to assess each connection method to reach the optimum path.

Therefore, the 2-Opt algorithm is applied to avoid premature stagnation of ACO and to not fall in local optimum. Figure 1 presents the architectural overview of AS-PSO-2Opt method.

2.4 Experimental Procedure

The principle of AS-PSO-2Opt is based on the ant number parameter that is a very important parameter in our algorithm and which is involved in the result. Consequently, we have conducted a comprehensive study on the choice of this parameter to find the best size that gives the optimal solution.

To justify the choice of ant number, several experiments are performed on a set of test. Therefore, we have chosen six test benches from TSPLIB Benchmark library [15] as eil51, berlin52, ST70, eil76, rat99 and eil101. The Best Known Solution (BKS) Given from TSPLIB of each test are respectively, 426, 7645, 675, 538, 1211 and 629.

The different experiments are carried out, firstly using a small number of ant, which is equal to five. After two tests are made using an ordinary ant number that are equal to 20 and 40. Finally, a test using a large ant number compared to the test bench. In this case, the selected ant number is equal to the city number of each test. Several Statistics are made and are presented in the next section.

3 Simulation and Results

To evaluate the proposed method, a statistical analysis was performed through 1000 executions using key test benches form TSPLIB Benchmark library [15].

Hardware specification used in the experiment is Core i5-2430M CPU 2.40 GHz, 4 GB memory and 500 GB for hard disk capacity. The development platform is the Matlab 2015a.

3.1 AS-PSO-2Opt Parameters Setting

AS-PSO-2Opt algorithm achieved good results in TSP applications using the combination parameters setting from PSO and ACO.

For PSO algorithm, it is executed 100 times with 10 particles. Several test benches as the eil51, berlin52, st70, eil76, rat99, kroA100 and eil101 are performed.

For ACO algorithm, Dorigo and Stutzle [16] choose fixed values of parameters ACO as $\alpha = 1, 2 <= \beta <= 5$, $\rho = 0.5$ and ant number is equal to the number of city. However, from the results obtained, we can conclude that these values are not the best. In [10], the authors select the values of parameters as $0 <= \alpha <= \beta <= 2$. For our algorithm, the best parameters values of ACO are variable in each iteration and they are optimized by PSO. The optimized parameters are limited to $0.5 <= \alpha <= 2$, $1 <= \beta <= 5$, $0 < \rho <= 0.5$ and for the ant number is selected after a comprehensive study.

3.2 Effects of Ants Number on AS-PSO-2Opt

To test the impact of ant number on the performance of the AS-PSO-2Opt algorithm and its selection in practical application, this paper analyzed and determined it through simulation. Different values of ant number are tested as 5, 20, 40 and ant number is equal to the number of city of each TSP test benches. The effects and simulation results of ant number on the performance of the AS-PSO-2Opt are shown in Figs. 2, 3 and 4 and Table 1 validates the choice of ant number parameter. Moreover, Table 1 shows the percentage error of these best solutions compared to the Best Known Solutions given in TSPLIB. The relative error is calculated according to Eq. (5), which is used in [7, 9–12].

$$Error(\%) = \frac{Average\ solution\ -\ Best\ known\ solution}{Best\ known\ solution} * 100. \tag{5}$$

Figures 2, 3 and 4 represent the effect of ant number on respectively berlin52, st70 and eil101 TSP instances. The three curves plot the ant number parameter according to the optimum course of each test. We can concluded, in all TSP instances, that the optimal solution is found when the ant number is equal to 20. For berlin52 test (Fig. 2), the optimal solution is equal to 7542 which is equal to Best Known Solution (BKS) and it is achieved when ant number is equal to 20. For st70 test (Fig. 3), the optimal solution is equal to 675 which is equal to Best Known Solution (BKS) and it is achieved when ant number is equal to 20. Finally, for eil101 test (Fig. 4), the optimal

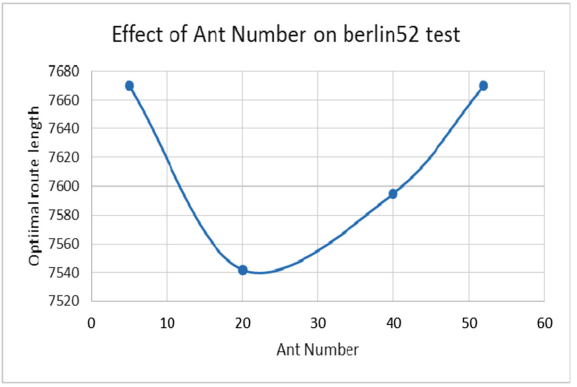


Fig. 2. Effects of ant number on berlin52 test performance.

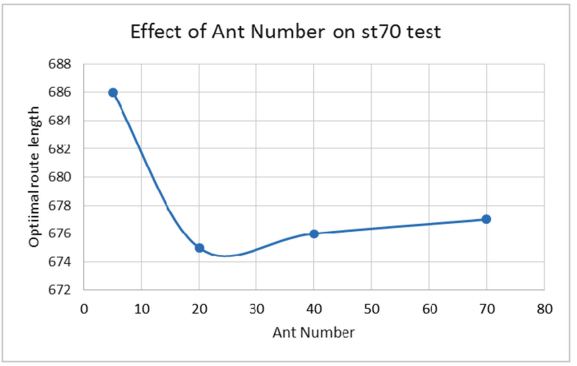


Fig. 3. Effects of ant number on st70 test performance.

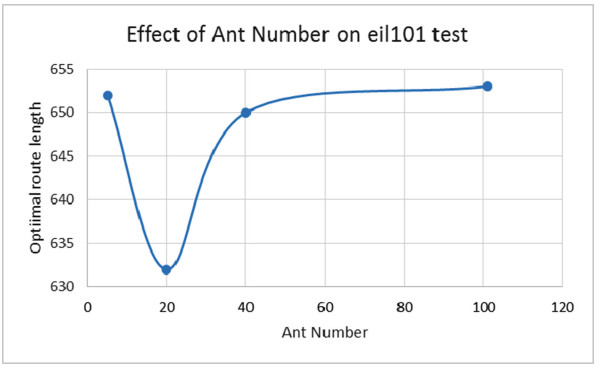


Fig. 4. Effects of ant number on eil101 test performance.

Table 1. The effects of ant numbers on the performance. **Best** is the best route length; **Worst** is the worst route length; **SD** is the standard deviation; **Error** (%) is percentage relative error; **Time** (s) is run time in seconds.

Problems	Values	Ant number = 5	Ant number = 20	Ant number = 40	Ant number = city number
eil51 426	Best	429	**428**	431	430
	Worst	475	467	482	492
	SD	9.36	8.63	79.74	11.11
	Error (%)	0.70	0.46	1.17	0.93
	Time (s)	465.45	547.67	793.90	885.15
berlin52 7542	Best	7670	**7542**	7596	7670
	Worst	9060	8606	9248	8602
	SD	263.37	316.24	309.28	211.19
	Error (%)	1.69	0	0.71	1.69
	Time (s)	297.76	182.56	442.21	634.55
st70 675	Best	686	**675**	676	677
	Worst	768	753	773	730
	SD	17.14	17.19	19.32	12.52
	Error (%)	1.62	0	0.14	0.29
	Time (s)	687.99	878.06	1749.83	5024.35
eil76 538	Best	546	**541**	545	555
	Worst	615	605	613	622
	SD	13.63	13.05	14.62	14.05
	Error (%)	1.48	0.55	1.30	3.15
	Time (s)	2612.04	1682.14	2016.80	3513.78
rat99 1211	Best	1235	**1234**	1252	1241
	Worst	615	1376	1398	1407
	SD	29.75	30.96	30.05	31.85
	Error (%)	1.98	1.89	3.38	2.4773
	Time (s)	1207.50	2550.36	2715.24	3190.94
eil101 629	Best	652	**632**	650	653
	Worst	723	702	715	712
	SD	13.85	14.57	12.88	12.49
	Error (%)	3.65	0.47	3.33	3.8156
	Time (s)	994.4719	2738.97	2685.07	2527.42

solution is equal to 632 which is closed to BKS that is 629 and it is achieved when ant number is equal to 20.

After execution of AS-PSO-2Opt algorithm, best solutions obtained and the optimum tour length are shown in Figs. 5 and 6, respectively. Therefore, Fig. 5 presents the optimum course of berlin52 test bench, which is equal to 7542 as the Best Known Solution (BKS). Figure 6 illustrates the optimum course of the eil101 test bench, which is equal to 632.

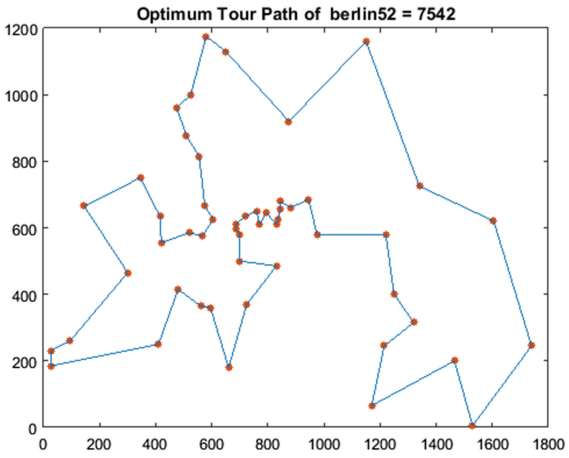


Fig. 5. Optimum tour of AS-PSO-2Opt for berlin52 test bench.

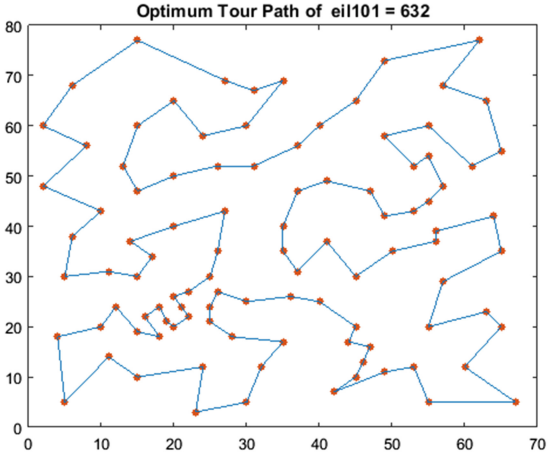


Fig. 6. Optimum tour of AS-PSO-2Opt for eil101 test bench.

Table 2 presents a comparison of our method using ant number equal to 20 with other research from literature. We can show that AS-PSO-2Opt gives interesting results compared to the BKS. The best results when our method performs better than others methods are presented on bold. Therefore, we can conclude that for two tests benches berlin52 and st70, the percentage error is zero because these problems give same solutions of the BKS.

Table 2. The computational results of AS-PSO-2Opt method and others methods from the literature. **Avg** is the average tour length.

Methods	Problem BKS	eil51 426	berlin52 7542	st70 675	eil76 538	rat99 1211	eil101 629
RABNET − TSP (2006) [17]	Avg.	438.70	8073.97	–	556.10	–	654.83
	SD	3.52	270.14		8.03		6.57
	Error (%)	2.98	7.05		3.36		4.11
Modified RABNET − TSP (2009) [18]	Avg.	437.47	7932.50	–	556.33	–	648.63
	SD	4.20	277.25		5.30		3.85
	Error (%)	2.69	5.18		3.41		3.12
ACO-2Opt (2012) [19]	Avg.	439.25	7556.58	–	–	–	672.37
	SD	–	–				–
	Error (%)	3.11	0.19				6.90
Hybrid ACO (2012) [20]	Avg.	431.20	7560.54	–	–	1241.33	–
	SD	2.00	67.48			9.60	
	Error (%)	1.22	0.23			2.5	
GA-Ant System (2012) [21]	Avg.	–	7634.00	–	542.00	–	–
	SD		–		–		
	Error (%)		1.22		0.74		
ACO-ABC (2015) [22]	Avg.	443.39	7544.37	700.58	557.98	–	683.39
	SD	5.25	0.00	7.51	4.10		6.56
	Error (%)	4.08	0.03	3.79	3.71		8.65
PSO-ACO-3Opt (α, β) (2015) [10]	Avg.	**426.45**	7543.20	678.20	**538.30**	**1227.40**	632.70
	SD	0.61	2.37	1.47	0.47	1.98	2.12
	Error (%)	0.11	0.02	0.47	0.06	1.35	0.59
AS-PSO-2Opt (α, β, ρ)	Avg.	428	**7542**	**675**	541	1234	**632**
	SD	8.63	316.24	17.19	13.05	30.96	14.57
	Error (%)	0.23	0.00	0.00	0.55	1.89	0.47

4 Conclusion and Perspectives

This paper studies the effects of swarm (ant) size on the performance of AS-PSO-2Opt method solving TSP. To demonstrate the choice of ant size, a selected set of test benches form TSPLIB are performed. In the experimental study, the ant number is selected from 5 to the city number of each TSP test benches. Experimental results showed that the optimal solution is achieved when the ant number is equal to 20.

Future investigations will be performed on the choice of PSO parameters and their effects on the performance of AS-PSO-2Opt metaheuristic method compared to others studies from literature.

Acknowledgments. The authors would like to acknowledge the financial support of this work by grants from General Direction of Scientific Research (DGRST), Tunisia, under the ARUB program.

References

1. Laporte, G.: The traveling salesman problem – an overview of exact and approximate algorithms. Eur. J. Oper. Res. **59**, 231–247 (1992)
2. Alba, E.: Parallel Metaheuristics: A New Class of Algorithms. Wiley-Interscience, New York (2005)
3. Glover, F.: Tabu search - part I. ORSA J. Comput. **1**(3), 190–206 (1989)
4. Grefenstette, J., Gopal, R., Rosmaita, B., Van Gucht, D.: Genetic algorithms for the traveling salesman problem. In: The First International Conference on Genetic Algorithms and Their Applications, pp. 160–168. Lawrence Erlbaum, NJ (1985)
5. Kennedy, J., Eberhart, R.: Particle swarm optimization. IEEE International Conference on Neural Networks **4**, 1942–1948 (1995)
6. Dorigo, M., Stutzle, T.: Ant Colony Optimization. Bradford Company, Scituate (2004)
7. Rokbani, N., Momasso, A.L., Alimi, A.M.: AS-PSO, ant supervised by PSO meta-heuristic with application to TSP. Proc. Eng. Technol. **4**, 148–152 (2013)
8. Rokbani, N., Abraham, A., Alimi, A.M.: Fuzzy ant supervised by PSO and simplified ant supervised PSO applied to TSP. In: The 13th International Conference on Hybrid Intelligent Systems, pp. 251–255 (2013)
9. Kefi, S., Rokbani, N., Krömer, P., Alimi, A.M.: A new ant supervised-PSO variant applied to traveling salesman problem. In: Abraham, A., Han, S.Y., Al-Sharhan, S.A., Liu, H. (eds.) Hybrid Intelligent Systems. AISC, vol. 420, pp. 87–101. Springer, Heidelberg (2016). doi:10.1007/978-3-319-27221-4_8
10. Mahia, M., Baykanb, Ö.K., Kodazb, H.: A new hybrid method based on particle swarm optimization, ant colony optimization and 3-Opt algorithms for traveling salesman problem. Appl. Soft Comput. **30**, 484–490 (2015)
11. Kefi, S., Rokbani, N., Kromer, P., Alimi, A.M.: Ant supervised by PSO and 2-Opt algorithm, AS-PSO-2Opt, applied to traveling salesman problem. In: IEEE International Conference on System Man and Cybernetics SMC (2016)
12. Chen, S.M., Chien, C.Y.: Solving the traveling salesman problem based on the genetic simulated annealing ant colony system with particle swarm optimization techniques. Expert Syst. Appl. **38**, 14439–14450 (2011)
13. Pasti, R., De Castro, L.N.: A neuro-immune network for solving the traveling salesman problem. In: The IEEE International Joint Conference on Neural Networks, pp. 3760–3766 (2006)
14. Kefi, S., Rokbani, N., Alimi, M.A.: Hybrid metaheuristic optimization based on ACO and standard PSO applied to traveling salesman problem. Int. J. Comput. Sci. Inf. Secur. (IJCSIS) **14**(7), 802–823 (2016)
15. Reinelt, G.: TSPLIB-a traveling salesman problem library. ORSA J. Comput. **3**, 376–384 (1991)
16. Dorigo, M., Stutzle, T.: Ant Colony Optimization, Massachusetts Institute of Technology (2004)
17. Pasti, R., Castro, L.N.D.: A neuro-immune network for solving the traveling salesman problem. In: The IEEE Inter Joint Conference on Neural Networks, pp. 3760–3766 (2006)
18. Masutti, T.A.S., De Castro, L.N.: A self-organizing neural network using ideas from the immune system to solve the traveling salesman problem. Inf. Sci. **179**, 1454–1468 (2009)
19. Jun-man, K., Yi, Z.: Application of an improved ant colony optimization on generalized traveling salesman problem. Energy Procedia **17**, 319–325 (2012)

20. Junqiang, W., Aijia, O.: A hybrid algorithm of ACO and delete-cross method for TSP. In: The IEEE International Conference on Industrial Control and Electronics Engineering, pp. 1694–1696 (2012)
21. Dong, G.F., Guo, W.W., Tickle, K.: Solving the traveling salesman problem using cooperative genetic ant systems. Expert Syst. Appl. **39**, 5006–5011 (2012)
22. Gunduz, M., Kiran, M.S., Ozceylan, E.: A hierarchic approach based on swarm intelligence to solve TSP. Turk. J. Electr. Eng. Comput. Sci. **23**, 103–117 (2015)

Improving Particle Swarm Optimization Using Co-Optimization of Particles and Acceleration Constants

Lin Wang, Bo Yang(✉), and Zhenxiang Chen

Shandong Provincial Key Laboratory of Network Based Intelligent Computing, University of Jinan, Jinan 250022, China
yangbo@ujn.edu.cn

Abstract. Particle swarm optimization exhibits effective performance for solving difficulties in real-world problems. However, the determination of acceleration constants, which influence the performance of PSO significantly and varies between different problems, is hard to tune. This paper presents a co-optimization strategy of particles and acceleration constants, which incorporates the optimization of parameters into the basic framework of optimization and extends the dimension of particles and embeds the acceleration constants at the additional part. Experiment results manifest that the proposed algorithm shows satisfactory performance.

Keywords: Particle swarm optimization · Co-optimization · Acceleration constants

1 Introduction

In many mathematical and scientific applications, optimization is a basic problem for optimizing parameters, structures, or solutions. Plenty of optimization methods have been proposed for decades, such as Powell's method [1], gradient descent etc. However, the landscape of many real-world applications are noisy, high-dimensional, and multimodal and result in the poor performance of traditional method. Biology inspired optimization, e.g., genetic algorithm [2], ant colony [3], was derived from the evolution of organisms or the behavior of animals. Particle swarm optimization (PSO) [4] is one of those approaches and, for real-world problems, exhibits effective and efficient performance in solving difficulties. It shows a series of successful cases in many fields, such as finance, energy, medicine, materials science, and remote sensing [5–12].

Every particle in particle swarm optimization is a vector and stands for a solution of target problem. In PSO, a swarm of particles move in searching space and are directed by two guides: *pbest* and *gbest*. *pbest* stands for the best particle position found by a particle in the history of searching. *gbest* stands

A. Abraham et al. (eds.), *Proceedings of the 16th International Conference on Hybrid Intelligent Systems (HIS 2016)*, Advances in Intelligent Systems and Computing 552,
DOI 10.1007/978-3-319-52941-7_57

for the best position found by the whole swarm up to now. In every iteration, the two guides are updated according to the new evaluation results of particles. The particle swarm optimization has received lots of success in avoiding local minimum and accelerating convergence. However, the determination of two key parameters φ_1 and φ_2, which influence the performance of PSO significantly and varies between different problems, is hard to tune.

The above-mentioned problem rises a challenge: *Is it possible for us to embed selection of parameters into the main framework yield better results for PSO?*

This paper presents a co-optimization of particles and acceleration constants (CPAC), which incorporates the optimization of parameters into the basic framework of optimization. It extends the dimension of particles and embeds the acceleration constants at the additional part. In optimization process, the method not only optimizes the solution vector, but also optimize the acceleration constants simultaneously. This method introduces the adaptive ability for PSO and enhance its effectiveness in solving real-world problems.

In the remaining parts of this paper, Sect. 2 reviews PSO and its applications in many fields. Section 3 detailed the proposed approach. Section 4 outlines experimental settings and discusses results. The last section makes the conclusion.

2 Related Works

As a biology-inspired algorithm, with the use of interaction and communication between individuals, PSO [4] enables the swarm to discover the global optimal region in a complex multi-modal searching space. Furthermore, in comparison with genetic algorithms, which were developed based on the evolution theory, it uses none of genetic mechanisms, such as crossover operator. The basic element of PSO is "particle" which size, mass, and volume are all zero. They fly around the searching space with a velocity following historical memory. The algorithm iteratively update particles' velocities and positions according to their memory which is defined by the best position found by population *gbest* and by particle itself *pbest*. The kernel equation for updating velocity and position is as follows

$$\begin{cases} v_{id}^d = v_{id}^d + \varphi_1 r(pbest_{id}^d - x_{id}^d) + \varphi_2 r(gbest_{id}^d - x_{id}^d) \\ x_{id}^d = x_{id}^d + v_{id}^d \end{cases} \tag{1}$$

where x, v, and d represent the position vector of particles, velocity vector of particles and dimension, respectively. The terms *pbest* and *gbest* represent the best position found by particle and population, respectively. r represents a uniformly distributed random positive number generated in interval [0, 1] and φ represent the acceleration constant.

This nature inspired approach has received lots of successful reports in many applications. In order to deal with reservoir operation, Zhang et al. [13] presented an adaptive PSO. For solving inference generated by pure parsimony, W Bin et al. [14] presented binary type PSO. For solving the load flow problems of

Algorithm 1. Algorithm Framework of PSO

Input: Population size PS, dimension of solution D, and acceleration constants φ_1, φ_2, maximum velocity $VMAX$
Output: The optimum result.

```
Initialization of particles;
while The maximum number of iteration has not been reached do
    Fitness evaluation according to particle's position x;
    Update pbest for each particle;
    Update gbest for all particles;
    for id=1 to PS do
        for d=1 to D do
            v_id^d = v_id^d + φ1 r(pbest_id^d − x_id^d) + φ2 r(gbest_id^d − x_id^d);
            v_id^d = min(VMAX^d, max(−VMAX^d, v_id^d));
            x_id^d = x_id^d + v_id^d;
        end
    end
end
Return the best result.
```

power system, E Davoodi et al. [15] combined quantum-behaved PSO with other algorithms.

A. Cervantes et al. [16] presented an adaptive Michigan particle swarm optimization and investigated its effectiveness in classification. Dezhi Li et al. [17] introduced particle swarm optimization to estimate parameters for fuzzy neural network and its use in system state forecasting. Hao Chen et al. [18] introduced quantum PSO for optimizing diagonal covariance matrix and tunable center vectors. For multi-objective optimization problems, Zhi-hui Zhan et al. [19] proposed a coevolutionary multiswarm particle swarm optimization. Qiang Lu et al. [20] adopts particle swarm optimization to make the determination for possible location of the odor source for robots.

In spite of the successful application stories of particle swarm optimization, the original version can still be further improved. One direction improves it by integrating the strategies of other methods to PSO. Mendes et al. [21] introduced a fully informed strategy to improve PSO by make all neighbors of particle to deliver information to it and updates velocity based on them. In this way, the updating of velocity is cumulative effect of all information in neighborhoods. It avoids the convergence to local optimal region by adding more source of information. J.J. Liang et al. [22] changed the topology framework of particle swarm optimization from static form to dynamic form. It divides the swarm into small swarms randomly and regroup them frequently. This approach makes the diversity increase and promotes the flow of information between swarms. They further proposed a comprehensive learning particle swarm optimizer (CLPSO) [23]. It independently optimizes vectors at each dimension and makes particle to receive information with a certain probability.

3 Methodology

In this methodology section, the procedure of co-optimization of particles and acceleration constants method is described in detail. In traditional particle swarm optimization, the determination of two key parameters φ_1 and φ_2, which influence the performance of PSO significantly and varies between different problems, is hard to tune. Therefore, we proposed a new method which embeds evolution of parameters into the main framework to improve the performance.

In this method, the dimension of traditional particle vector is extended from D to $3D$ in which additional dimensions are used for the evolution of acceleration constants. The evolution of acceleration constants also make use of the

Algorithm 2. Algorithm Framework of CPAC

Input: Population size PS, dimension of solution D, and acceleration constants φ, maximum velocity $VMAX$
Output: The optimum result.

```
Initialization of particles;
while The maximum number of iteration has not been reached do
    Fitness evaluation according to particle's position x;
    Update pbest for each particle;
    Update gbest for all particles;
    for id=1 to PS do
        for d=1 to 2D do
            if d ≤ D then
                φ^d_{1 id} = f(x^{D+d}_{id});
            end
            else
                φ^{d−D}_{2 id} = f(x^{D+d}_{id});
            end
        end
    end
    for id=1 to PS do
        for d=1 to 3D do
            if d ≤ D then
                v^d_{id} = v^d_{id} + φ^d_{1 id} r(pbest^d_{id} − x^d_{id}) + φ^d_{2 id} r(gbest^d_{id} − x^d_{id});
            end
            else
                v^d_{id} = v^d_{id} + φr(pbest^d_{id} − x^d_{id}) + φr(gbest^d_{id} − x^d_{id});
            end
            v^d_{id} = min(VMAX^d, max(−VMAX^d, v^d_{id}));
            x^d_{id} = x^d_{id} + v^d_{id};
        end
    end
end
Return the best result.
```

framework of particle swarm optimization with fixed acceleration constant φ. The new velocity equation is defined as

$$v_{id}^d = v_{id}^d + \varphi_{1\ id}^d r(pbest_{id}^d - x_{id}^d) + \varphi_{2\ id}^d r(gbest_{id}^d - x_{id}^d) \tag{2}$$

where $\varphi_{1\ id}^d$ and $\varphi_{2\ id}^d$ is encoded in the latter part of extended particle id. In order to restrict the range of acceleration constant to a reasonable interval and accelerate convergence, the original value of evolved acceleration constant is remapped to

$$f(x) = \alpha \frac{1 - e^{-x}}{1 + e^{-x}} + \beta \tag{3}$$

where α represents the maximum value of acceleration constant. In this way, the acceleration constant for each dimension can be automatically determined and fits given problem. Algorithm 2 illustrates the details of proposed method.

3.1 Experiments

Benchmark Functions and Compared Methods. Sixteen benchmark functions [24,25] are used to verify the effectiveness of CPAC by comparing it with other methods. The following list exhibits the definition and features of benchmark functions.

- $F_1(x) = \sum_{i=1}^{D} {x_i}^2$. Search Range is $[-100, 100]^D$. Best value is 0.
- $F_2(x) = \sum_{i=1}^{D} \left(\sum_{j=1}^{i} x_j \right)^2$. Search Range is $[-100, 100]^D$. Best value is 0.
- $F_3(x) = \sum_{i=1}^{D} i{x_i}^4 + random[0, 1)$. Search Range is $[-1.28, 1.28]^D$. Best value is 0.
- $F_4(x) = \sum_{i=1}^{D} (\lfloor x_i + 0.5 \rfloor)^2$. Search Range is $[-100, 100]^D$. Best value is 0.
- $F_5(x) = \sum_{i=1}^{D-1} \left(100 \left({x_i}^2 - x_{i+1} \right)^2 + (x_i - 1)^2 \right)$. Search Range is $[-2, 2]^D$. Best value is 0.
- $F_6(x) = -20e^{-0.2\sqrt{\frac{1}{D}\sum_{i=1}^{D} x_i^2}} - e^{\frac{1}{D}\sum_{i=1}^{D} \cos(2\pi x_i)} + 20 + e$. Search Range is $[-32, 32]^D$. Best value is 0.
- $F_7(x) = \sum_{i=1}^{D} \frac{x_i^2}{4000} - \prod_{i=1}^{D} \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$. Search Range is $[-600, 600]^D$. Best value is 0.
- $F_8(x) = \sum_{i=1}^{D} \left(x_i^2 - 10\cos(2\pi x_i) + 10 \right)$. Search Range is $[-5, 5]^D$. Best value is 0.

– $\begin{cases} F_9(x) = \sum_{i=1}^{D} \left(y_i^2 - 10\cos(2\pi y_i) + 10\right) \\ y_i = \begin{cases} x_i & |x_i| < \frac{1}{2} \\ \frac{round(2x_i)}{2} & |x_i| \geq \frac{1}{2} \end{cases} \quad (i = 1, 2, \cdots, D). \end{cases}$

Search Range is$[-5, 5]^D$. Best value is 0.

– $\begin{cases} F_{10}(x) = \sum_{i=1}^{D-1} \left(100\left(z_i^2 - z_{i+1}\right)^2 + (z_i - 1)^2\right) + 390 \\ z = x - o + 1. \end{cases}$

Search Range is $[-100, 100]^D$. Best value is 390.

– $\begin{cases} F_{11}(x) = \sum_{i=1}^{D} \frac{z_i^2}{4000} - \prod_{i=1}^{D} \cos(\frac{z_i}{\sqrt{i}}) + 1 - 180 \\ z = (x - o) * M. \end{cases}$

Search Range is $[0, 600]^D$. Best value is -180.

– $\begin{cases} F_{12}(x) = \sum_{i=1}^{D} (z_i^2 - 10\cos(2\pi z_i) + 10) - 330 \\ z = x - o. \end{cases}$

Search Range is $[-5, 5]^D$. Best value is -330.

– $\begin{cases} F_{13}(x) = \sum_{i=1}^{D} (z_i^2 - 10\cos(2\pi z_i) + 10) - 330 \\ z = (x - o) * M. \end{cases}$

Search Range is $[-5, 5]^D$. Best value is -330.

– $\begin{cases} F_{14}(x) = \sum_{i=1}^{D} (\sum_{k=0}^{k\max} \left[a^k \cos(2\pi b^k (z_i + 0.5))\right]) \\ \qquad\quad - \sum_{i=1}^{D} (\sum_{k=0}^{k\max} \left[a^k \cos(2\pi b^k * 0.5)\right]) + 90 \\ a = 0.5, b = 3, k\max = 20 \\ z = (x - o) * M. \end{cases}$

Search Range is $[-0.5, 0.5]^D$. Best value is 90.

– $\begin{cases} F_{15}(x) = F_7(F_5(z_1, z_2)) + F_7(F_5(z_2, z_3)) + \cdots \\ \qquad\quad + F_7(F_5(z_{D-1}, z_D)) + F_7(F_5(z_D, z_1)) - 130 \\ z = x - o + 1. \end{cases}$

Search Range is $[-3, 1]^D$. Best value is -130.

– $\begin{cases} F_{16}(x) = F(z_1, z_2) + F(z_2, z_3) + \cdots \\ \qquad\qquad +F(z_{D-1}, z_D) + F(z_D, z_1) - 300 \\ z = (x - o) * M \end{cases}$
Search Range is $[-100, 100]^D$. Best value is -300.

The following evolutionary algorithms are used for comparison. Genetic algorithm (GA) which is one of the most important basic evolutionary algorithms [2]. The traditional global inertia weighted particle swarm optimization (GPSO) [4]. Furthermore, its local version (LPSO) [26] was also adopted, which uses ring topology as the neighborhood method. Besides the partially informed particle swarm optimization, a fully informed version (FIPS) [21] were also used for comparison. Two topologies are used, including ring topology with self removed from neighborhood (UR) and square topology with self removed from neighborhood (US). A dynamic multi-swarm particle swarm optimization [22], which reorganizes topologies iteratively, is also adopted for comparison (DMS-PSO). Moreover, an improved method which uses comprehensive learning approach optimizing each dimension independently (CLPSO) [23] is used in experiment.

Parameters Settings. In order to compare different method fairly, the common parameters are set to the same value, where maximum fitness evaluation is set to $2E + 5$, population size is set to 64. All approaches were run 30 times for each benchmark function to report statistical results. The results of compared method are from Refs. [27,28]. For CPAC, the φ was set to 1.8, α was set to 0.2, and β was set to 1.8.

Results and Discussions. All methods are compared in the same machine using Matlab environment. Table 1 and Table 2 report the mean and SD results of sixteen benchmark functions on ten and thirty dimensions, respectively. It can be observed from tables that CPAC achieves the best results on many functions. Especially, the performance of CPAC reaches better results than the others for unimodal problems in low dimensional case. However, it performs worse, in low dimensional case, for multimodal functions. Noticeably, the trend is reversed when the dimension is increased to thirty. CPAC performs better than the others for multimodal functions but performs worse for unimodal functions. This trend indicates that the CPAC method is suitable for solving difficult problems which landscapes are complex and dimensions are high. This new method overcomes the difficulty that particle swarm optimization is facing in determining acceleration constants for different problems. It enables the adaption of parameters for each dimension and improves the performance.

Table 1. Report of mean of all methods for ten dimensional benchmark functions. The results of compared method are from Refs. [27,28].

	F_1	F_2	F_3	F_4	F_5	F_6	F_7	F_8
CPAC	**8.90E-126**	**1.08E-50**	**3.06E-04**	**0.00E+00**	1.84E+00	2.55E-15	5.68E-02	1.49E+00
CLPSO	3.46E-102	4.42E-14	7.57E-04	**0.00E+00**	3.44E+00	2.66E-15	2.72E-02	1.76E+00
DMSPSO	3.00E-19	1.40E-02	1.44E-03	**0.00E+00**	4.23E+00	1.65E-10	1.44E-01	3.69E+00
GPSO	6.93E-89	5.44E-31	5.14E-04	**0.00E+00**	2.19E+00	2.66E-15	6.04E-02	1.26E+00
LPSO	1.90E-40	5.63E-09	1.22E-03	**0.00E+00**	3.73E+00	2.66E-15	3.24E-02	2.04E+00
FIPS(UR)	5.22E-28	2.13E-10	6.70E-04	**0.00E+00**	2.59E+00	3.37E-11	1.80E-02	2.67E-01
FIPS(US)	9.68E-76	5.60E-28	4.61E-04	**0.00E+00**	3.09E-01	**1.01E-15**	2.94E-03	**4.57E-02**
GA	4.06E-11	1.30E-09	9.22E-02	**0.00E+00**	**7.65E-02**	7.63E-06	**5.05E-12**	5.31E-01
	F_9	F_{10}	F_{11}	F_{12}	F_{13}	F_{14}	F_{15}	F_{16}
CPAC	1.37E+00	4.61E+00	**1.99E-01**	1.49E+00	1.13E+01	**1.95E+00**	5.52E-01	2.56E+00
CLPSO	2.07E+00	1.15E+01	1.27E+03	1.99E+00	1.40E+01	4.11E+00	**3.99E-01**	2.63E+00
DMSPSO	4.90E+00	1.19E+01	2.65E-01	4.20E+00	8.19E+00	2.22E+00	8.54E-01	2.15E+00
GPSO	**6.67E-02**	8.24E+00	3.49E+02	6.88E-01	1.19E+00	1.06E+02	4.40E+02	6.70E-01
LPSO	1.93E+00	1.46E+00	2.51E+00	**2.97E-02**	2.10E+00	8.96E+01	4.49E+00	**6.06E-01**
FIPS(UR)	3.58E+00	3.57E+00	4.13E+01	3.72E-01	**2.38E-01**	1.50E+01	4.98E+03	1.47E+00
FIPS(US)	2.30E+00	**7.41E-01**	9.35E-01	1.60E-01	1.45E+01	4.05E+00	1.81E+00	2.56E+00
GA	4.77E+00	1.67E+01	1.67E+01	1.93E+02	3.45E+00	1.16E+02	1.19E+03	2.77E+00

Table 2. Report of mean of all methods of all methods for thirty dimensional benchmark functions. The results of compared method are from Refs. [27,28].

	F_1	F_2	F_3	F_4	F_5	F_6	F_7	F_8
CPAC	4.30E-11	4.32E+00	5.51E-03	2.67E-01	**2.18E+01**	1.38E-01	1.51E-02	3.07E+01
CLPSO	7.21E-21	1.79E+02	**3.55E-03**	**0.00E+00**	2.56E+01	1.87E-11	**0.00E+00**	**2.26E+00**
DMSPSO	2.34E-01	1.40E+03	5.50E-02	3.93E+00	2.72E+01	6.88E-01	4.71E-01	6.54E+01
GPSO	1.04E-21	1.56E+01	9.83E-03	**0.00E+00**	2.43E+01	5.85E-12	1.48E-02	2.81E+01
LPSO	5.13E-08	4.25E+02	2.54E-02	**0.00E+00**	2.52E+01	1.05E-04	9.25E-04	3.62E+01
FIPS(UR)	3.44E-05	3.96E+03	1.13E-02	**0.00E+00**	2.56E+01	1.21E-03	2.97E-03	1.02E+02
FIPS(US)	**3.53E-42**	**6.15E-02**	4.11E-03	**0.00E+00**	2.28E+01	**4.20E-15**	4.59E-05	2.51E+01
GA	6.40E-09	2.00E-02	1.11E+00	2.00E-01	2.23E+01	5.95E-05	3.50E-10	4.94E+00
	F_9	F_{10}	F_{11}	F_{12}	F_{13}	F_{14}	F_{15}	F_{16}
CPAC	2.27E+01	8.03E+01	**1.71E-01**	3.19E+01	**7.15E+01**	**1.90E+01**	2.88E+00	**1.22E+01**
CLPSO	**5.20E+00**	8.13E+01	4.70E+03	4.94E+00	9.44E+01	3.21E+01	**1.73E+00**	1.30E+01
DMSPSO	4.84E+01	5.36E+03	1.20E+00	6.77E+01	1.13E+02	2.71E+01	9.35E+00	1.23E+01
GPSO	1.66E+01	1.81E+02	4.41E+02	**1.94E-02**	7.82E+01	1.95E+01	2.81E+00	1.24E+01
LPSO	3.55E+01	6.25E+01	1.86E+02	4.79E-02	1.31E+02	2.09E+01	5.10E+00	1.25E+01
FIPS(UR)	9.57E+01	3.50E+02	1.42E+04	2.09E+00	2.01E+02	3.99E+01	1.65E+01	1.40E+01
FIPS(US)	3.34E+01	**5.18E+01**	1.96E+01	8.37E+01	1.69E+02	3.48E+01	2.67E+01	1.26E+01
GA	2.24E+01	3.17E+02	2.56E+02	3.54E+03	5.35E+02	3.46E+01	2.38E+01	1.46E+01

4 Conclusions

In this paper, a co-optimization strategy of particles and acceleration constants is proposed for improving the performance of particle swarm optimization. It incorporates the optimization of parameters into main framework and embeds

the acceleration constants at the additional part of particles. To evaluate its performance, sixteen benchmark functions are selected to compare it with other methods. Experiments manifest that the performance of CPAC reaches better results than the others in many benchmark functions. In future works, the proposed CPAC method will be applied for real-world applications to further verify its effectiveness.

Acknowledgement. This work was supported by National Natural Science Foundation of China under Grant No. 61573166, No. 61572230, No. 61373054, No. 61472164, No. 81301298, No.61302128, No. 61472163. Shandong Provincial Natural Science Foundation, China, under Grant ZR2015JL025. Science and technology project of Shandong Province under Grant No. 2015GGX101025.

References

1. Powell, M.J.D.: An efficient method for finding the minimum of a function of several variables without calculating derivatives. Comput. J. **7**(2), 155–162 (1964)
2. Holland, J.H.: Adaptation in Natural and Artificial Systems. University of Michigan Press, Ann Arbor (1975)
3. Dorigo, M.: Optimization, "Learning and Natural Algorithms". Ph.D. thesis, Politecnico di Milano, Italy (1992)
4. Kennedy, J., Eberhart, R.C.: A new optimizer using paritcle swarm theory. In: Proceedings of the Sixth International Symposium on Micromachine and Human Science, pp. 39–43 (1995)
5. Ciurana, J., Arias, G., Ozel, T.: Neural network modeling and particle swarm optimization (PSO) of process parameters in pulsed laser micromachining of hardened AISI H13 steel. Mater. Manuf. Processes **24**(3), 358–368 (2009)
6. Eslami, M., Shareef, H., Mohamed, A.: Power system stabilizer design using hybrid multi-objective particle swarm optimization with chaos. J. Central South Univ. Technol. **18**(5), 1579–1588 (2011)
7. Hung, J.-C.: Adaptive Fuzzy-GARCH model applied to forecasting the volatility of stock markets using particle swarm optimization. Inf. Sci. **181**(20), 4673–4683 (2011)
8. del Valle, Y., Venayagamoorthy, G.K., Mohagheghi, S., et al.: Particle swarm optimization: basic concepts, variants and applications in power systems. IEEE Trans. Evol. Comput. **12**(2), 171–195 (2008)
9. Zainud-Deen, S.H., Hassen, W.M., Ali, E.M., et al.: Breast cancer detection using a hybrid finite difference frequency domain and particle swarm optimization techniques. Prog. Electromagnet. Res. B **3**, 35–46 (2008)
10. Wang, L., Yang, B., Chen, Y., Zhang, X., Orchard, J.: doi:10.1109/TNNLS.2016.2580570 (in press)
11. Lim, T.S., Koo, V.C., Ewe, H.T., et al.: A SAR autofocus algorithm based on particle swarm optimization. Prog. Electromagnet. Res. B **1**, 159–176 (2008)
12. Wang, L., Yang, B., Abraham, A.: Distilling middle-age cement hydration kinetics from observed data using phased hybrid evolution. Soft. Comput. **20**, 3637–3656 (2016)
13. Zhang, Z.B., Jiang, Y.Z., Zhang, S.H., Geng, S.M., Wang, H., Sang, G.Q.: An adaptive particle swarm optimization algorithm for reservoir operation optimization. Appl. Soft Comput. **18**, 167–177 (2014)

14. Bin, W., Jing, Z.: Haplotype inference using a novel binary particle swarm optimization algorithm. Appl. Soft Comput. **21**, 415–422 (2014)
15. Davoodi, E., Hagh, M.T., Zadeh, S.G.: A hybrid Improved Quantum-behaved Particle Swarm Optimization-Simplex method (IQPSOS) to solve power system load flow problems. Appl. Soft Comput. **21**, 171–179 (2013)
16. Cervantes, A., Galvan, I.M., Isasi, P.: AMPSO: a new particle swarm method for nearest neighborhood classification. IEEE Trans. Cybern. **39**(5), 1082–1091 (2009)
17. Li, D., Wang, W., Ismail, F.: Fuzzy neural network technique for system state forecasting. IEEE Trans. Cybernet. **43**(5), 1484–1494 (2013)
18. Hao Chen, Y., Gong, X.H.: Online modeling with tunable RBF network. IEEE Trans. Cybernet. **43**(3), 935–947 (2013)
19. Zhan, Z.-H., Li, J., Cao, J., Zhang, J., et al.: Multiple populations for multiple objectives: a coevolutionary technique for solving multiobjective optimization problems. IEEE Trans. Cybernet. **43**(2) (2013)
20. Qiang, L., Liu, S., Xie, X., Wang, J.: Decision making and finite-time motion control for a group of robots. IEEE Trans. Cybernet. **43**(2), 738–750 (2013)
21. Mendes, R., Kennedy, J., Neves, J.: The fully informed particle swarm: simpler, maybe better. IEEE Trans. Evol. Comput. **8**(3), 204–210 (2004)
22. Liang, J.J., Suganthan, P.N.: Dynamic multi-swarm particle swarm optimizer. In: Proceedings of the Swarm Intelligence Symposium, pp. 124–129 (2005)
23. Liang, J.J., Qin, A.K., Suganthan, P.N., Baskar, S.: Comprehensive learning particle swarm optimizer for global optimization of multimodal functions. IEEE Trans. Evol. Comput. **10**(3), 281–295 (2006)
24. Yao, X., Liu, Y., Lin, G.M.: Evolutionary programming made faster. IEEE Trans. Evol. Comput. **3**(2), 82–102 (1999)
25. Suganthan, P.N., Hansen, N., Liang, J.J., Deb, K., Chen, Y.P., Auger, A., Tiwari, S.: Problem definitions and evaluation criteria for the CEC 2005 special session on real-parameter optimization. Technical report, Nanyang Technological University, Singapore and KanGAL Report Number 2005005 (2005)
26. Kennedy, J., Mendes, R.: Population structure and particle swarm performance. In: Proceedings of IEEE Congress on Evolutionary Computation, pp. 1671–1676 (2002)
27. Wang, L., Yang, B., Chen, Y.: Imrpoving particle swarm optimization using multilayer searching strategy. Inf. Sci. **274**, 70–94 (2014)
28. Wang, L., Yang, B., Orchard, J.: Imrpoving Particle Swarm Optimization using Multi-Layer Searching Strategy. Appl. Soft Comput. **48**, 584–96 (2016)

Spread Control for Huge Data Fuzzy Learning

Monia Tlili(✉), Tarek M. Hamdani, and Adel M. Alimi

REGIM Lab: REsearch Groups in Intelligent Machines, National Engineering School of Sfax (ENIS), University of Sfax, BP 1173, 3038 Sfax, Tunisia
{monia.tlili,tarek.hamdani,adel.alimi}@ieee.org

Abstract. The control of Growing Self Organizing Maps (GSOM) algorithms presents a serious issue with huge data learning. In conjunction with the growing threshold (GT), the spread factor (SF) is used as a controlling measure of the map size during the growing process. The effect of the spread factor in fuzzy learning with Fuzzy Multilevel Interior GSOMs (FMIG) algorithm is investigated. Further analysis is conducted on very large data in order to demonstrate the spread control of data distribution with FMIG learning in comparison with Multilevel Interior Growing SOM (MIGSOM), GSOM, Fuzzy Kohonen Clustering Network (FKCN) and fuzzy GSOM. Therefore, the aim of this paper is to study the effect of the spread factor values on the map structure in term of quantization error, topology preservation and dead units. Experimental studies with huge synthetic and real datasets are fulfilled at different spread factor values for the advertised algorithms.

Keywords: Fuzzy learning · Spread factor · Threshold growing · Multilevel interior growing self-organizing maps · Quantization · Topology

1 Introduction

With the evolution of scientific areas nowadays, the data used are becoming increasingly huge, inform a growing number of variables and cover big fields. In fact, real life problems handle data that become large and complex [3]. Learning such huge data is a useful tool in its visualizing. The Self-Organizing Map (SOM) (Kohonen [14]) is an unsupervised learning model used to explore and visualize linear and nonlinear relationships in huge data. In literature researches, SOM is successfully a static technique used in visualization of rich and complex phenomena such as pattern recognition tasks involving different criteria especially in clustering mechanism [17, 18, 20]. Different variants of SOM were presented in order to ameliorate the quality of the map such as Growing SOM (GSOM) [15], Multilevel Interior Growing Self-Organizing Maps (MIGSOM) [5], Fuzzy Kohonen Clustering Network (FKCN) [9], and our developed Fuzzy Multilevel Interior GSOMs (FMIG) [1].

Controlling the growing of the map during the learning process is a critical issue [2, 20]. The static SOM algorithms do not involve a system for identifying the size and the spread level of a feature map [4, 11, 16]. Thus, the SOM data learning can only refer to the predefined length and width of the grid to develop the map structure. Dynamic variants of SOM were developed in order to successfully solve the problem

A. Abraham et al. (eds.), *Proceedings of the 16th International Conference on Hybrid Intelligent Systems (HIS 2016)*, Advances in Intelligent Systems and Computing 552,
DOI 10.1007/978-3-319-52941-7_58

of predefining the suitable size of the map [15, 18, 21]. In fact, growing SOM variants start the learning process with a minimum number of nodes and then dynamically add new nodes to develop the map structure.

The spread factor (SF) is a parameter of map size measure and control [3, 6]. SF provides the ability to control the growth of the map which is unique to the GSOM algorithms to achieve progressive learning of a dataset. The spread factor is independent of the size of the dataset. Hence, SF provides a measure of the level of growing across different datasets. This parameter can be assigned values between 0 and 1. SF is used to calculate the growing threshold parameter (GT). The learning process uses such parameter GT for initiating node generation. A large GT value will result in less spread out map with fewer nodes, and a low GT will produce a well-spread map [7, 15]. The main goal of this paper is to provide a study of the spread factor effect on the map growing with the new fuzzy learning method FMIG [1, 2] in comparison with MIGSOM [5], FKCN [9], GSOM [15], and fuzzy GSOM developed by Tlili M. et al. [1], in term of quantization error, topologic error and dead units [1, 4, 6, 19].

FMIG approach is able to give the best distribution of data and match the aims of preserving the topology and minimizing the quantization error [1, 3]. FMIG has the capacity to interpret the overall geometric form of large data by matching between structures of the output maps and the input patterns structure. A new form of Growing Threshold is presented to control the development of the FMIG network in the case of large data taking into account the complicated computing process.

This paper is organized as following: Sect. 2 reviews the FMIG approach and the growing threshold function. Section 3 presents the study of the spread factor effect on the iterative learning process. Then, Sect. 4 presents an experimental study of the spread factor influence on learning synthetic and real datasets. Finally, Sect. 5 summarises and concludes our remarks.

2 An Overview on Fuzzy Multilevel Interior GSOMs (FMIG) Approach

Fuzzy Multilevel Interior GSOMs (FMIG) presents a learning approach based on the MIGSOM algorithm [5] and FKCN algorithm [9]. FKCN combines between the robustness of MIGSOM with large data learning and the fuzzy aspect of FKCN training. FMIG shows two interesting advantages. First, it produces a well distributed neural map of a given data by finding out the topology corresponding to the form of inputs [1]. Second, it gives the minimum value of error quantization providing the best cluster structure [1].

FMIG algorithm is composed of three steps: initialization phase, growing phase and smoothing phase.

1. ***Initialisation***

- Initialize the pre-defined values of the Spread Factor (SF) used to control the growth of the map taken in [0, 1], and m_0 the fuzzifier factor.

- According to the number of points in the training data set (n) and the dimension of its characteristics (D), the growing threshold gives an identical image of the specific data aspects. our New Growing Threshold (NGT) is defined as follows) as [1]:

$$\mathrm{NGT} = \frac{(-\ln(D) \times \ln(\mathrm{SF}))}{\ln(n)}. \tag{1}$$

- Initialize randomly the map grid with (2 × 2) or (3 × 3) c_0 nodes and the weight vectors $v_0 = (v_{1,0}, v_{2,0}, v_{c0,0})$.
- Determine t_{max} the limit of iterations and initialize randomly the membership matrix, with n the number of data inputs:

$$U = \left|u_{ij}\right|_{(\mathrm{i}=1\ldots \mathrm{c0}, \mathrm{j}=1\ldots \mathrm{n})}. \tag{2}$$

2. ***Growing Process***
For t = 1… t_{max}:

- Calculate the fuzzifier parameter m_t as shown:

$$m_t = m_0 - t \times \Delta m. \tag{3}$$

$$\Delta m = \frac{(m_0 - 1)}{t_{\max}}. \tag{4}$$

- Update the membership degree $u_{ik,t}$ for the iteration t using the Euclidean distance between the input pattern $x_k$and the weight vector v_i, with c_t is the number of weight vectors at the iteration t:

$$u_{ik,t} = \left[\sum_{j=1}^{c_t}\left[\frac{(x_k - v_i)}{(x_k - v_j)}\right]^{2/(m_t-1)}\right]^{-1}. \tag{5}$$

- Compute the learning rate $\alpha_{ik,t}$ as defined by:

$$\alpha_{ik,t} = \left(u_{ik,t}\right)^{m_t}. \tag{6}$$

- Update the weight vectors as follows:

$$v_{i,t} = v_{i,t-1} + \frac{\sum_{k=1}^{n} \alpha_{ik,t}\left(x_k - v_{i,t-1}\right)}{\sum_{s=1}^{n} \alpha_{is,t}} \tag{7}$$

- Determine the quantization error of each node i as [1]:

$$QE_i = \sum_{j}^{nbu} \left\| x_j - vi \right\|. \tag{8}$$

With *nbu*, the number of units mapped by the node *i*.

- Calculate the error of each node. Compute the node presenting the highest quantization error QE_{max} called node q with k units mapped in as follows:

$$E_{rr} = \sum_{j}^{k} \left\| x_j - w_q \right\|. \tag{9}$$

-

$$\text{If } QE_{max} > NGT \text{ then generate new nodes from } q. \tag{10}$$

3. ***Smoothing Process***

- Initialize the limit of smoothing iterations t_{max}.
- Train the map as the growing process without adding new nodes as [1].

3 Why Controlling the Spread of Huge Data Learning?

Huge data presents large dimension in:

- the size of the data or the number of vectors,
- the number of clusters,
- the dimensionality or the number of data features.

To train very large datasets, it is imperative to correctly initialize growing control parameters in order to produce a well-distributed map. This resulting map must be a faithful image that represents the trained dataset respecting its huge dimensionality (size, number of clusters and number of features). In addition, the criteria of the dataset such as compactness and overlap should be well-preserved by the created map. For these reasons, the current studies focus on the spread factor variations and its impact on the learning process results.

The spread factor must be defined in order to allow the data analyst to control the development of the map. SF is used in the first step of the training process to calculate the growing threshold. SF will act as an index where the choice of its value has a direct influence on GT value for initiating new node generation. As GT corresponds to the stop condition of the growing process (Eq. (10)), an important GT value will provide less distribution (spread-out) map. However, a small GT value will generate a well-distributed map. The new growing threshold as we have defined in FMIG approach takes into account the size of the training dataset (n) as well as the dimension of its characteristics (D).

Example: For SP = 0.8
Data1 (D = 4, n = 1000); Data2 (D = 4, n = 50000);

$$\mathrm{NGT}(\mathrm{Data1}) = 0.044$$
$$\mathrm{NGT}(\mathrm{Data2}) = 0.028$$

With the use of GT function as defined in [5]:

$$GT = -ln(D)xln(SP)$$
$$\mathrm{GT}(\mathrm{Data1}) = \mathrm{GT}(\mathrm{Data2}) = 0.309$$

GT is the same for Data1 and Data2 without considering their different sizes. That is why the stop condition in the iterative process of GSOM algorithms, is the same for datasets presenting the identical dimension D and different sizes. Therefore, resulting maps do not match the inputs.

NGT as defined by Eq. (1) gives different values. Since Data2 (50000 points) is very large compared to Data1 (1000 points), NGT(Data2) is lower than NGT(Data1). This test confirms that in iterative processes, the number of iterations is important to evaluate the results because it has the probability to give the global solution of the problem. Indeed, the iterative process executes the maximum of iterations to produce the optimal result when the stop condition is chosen correctly. The spread factor effect will be tested by different SF values in the training process of FMIG, MIGSOM, GSOM, Fuzzy GSOM and FKCN algorithms. Therefore, the SF value that provides the best distribution of the map in term of quantization error, topologic error and dead units, gives an identical image of the specific data aspects.

4 Experimental Studies

In this section, experimental studies are conducted to investigate the effect of the spread factor on the control of the network developing. GSOMs algorithms are executed: FMIG, MIGSOM, GSOM, Fuzzy GSOM and FKCN in the experimental environment presented in Table 1.

Table 1 Experimental parameters

Distance function	Fuzzy factor m0	Max growing iter tmax	Max smoothing iter tsmax
Euclidean	2	500	100

4.1 Experimental Setting

The evaluation of the spread factor control is made in term of quantization and topologic errors, and dead units presented as follows:

- The Final Quantization Error: FQE is computed as the normalized average distance between each input data and its BMU [5].

$$FQE = \begin{cases} 1 \text{ if no data point matches the unit} \\ \frac{1}{c_t}\sum_{i=1}^{c_t}\left[\frac{\frac{1}{nbu}\sum_{j=1}^{nbu}\left\|x_j - v_{BMUt}\right\|}{norm(v_i)}\right] \end{cases} \tag{11}$$

With v_{BMUt} is the weight vector of the BMU at the iteration t.

- Topologic Error: The Final Topologic Error FTE is computed identically to [5] as follows:

$$FTE = \frac{1}{c_t}\sum_{i=1}^{c_t} mu(v_i). \tag{12}$$

$mu(v_i)$ takes the values 1 if the first and the second Best Matching Units (BMUs) of v_i are adjacent, 0 otherwise.

- Dead units: the number of nodes which are never organized by the input data. In other words in the training process, some neurons fail to be associated with any input vectors. It is convenient to encounter dead units in the aim of compute the unlabelled nodes that do not match any input vectors [4, 7, 8, 11].

4.2 Data Sets Description

We performed a sensitivity analysis in synthetic data sets with different dimensions (Dataset1 and Dataset2) and real data (DNA [21], Letter [21] and NE [22]). Table 2 describes the characteristics of these datasets in term of size, dimension or number of features, number of clusters and aspects (overlap and noise).

Table 2 Datasets description

Database	Nb points	Dimension	Nb clusters	Aspect
Dataset1	1600	2	4	Overlapping
Dataset2	1300	2	13	Noisy
DNA	2000	180	3	Noise + overlap
Letter	20000	16	26	Overlapping
NE	50000	2	3	Overlapping

4.3 Experiments and Discussion

This work was performed to study the effect of the spread factor changes on the learning results regarding the quantization and topologic errors and the dead units.

We executed the algorithms FMIG, MIGSOM, GSOM, Fuzzy GSOM and FKCN in the same experimental environment described in Table 1. In this analysis, a spread factor value was given at the beginning of the learning process, and then we gradually increased SF for further observations of the different parameters. The spread factor takes values between 0.2 and 0.9 and is independent of the number of dimensions in the data. For the same spread factor, we compare the results of different algorithms with a number of attributes.

Table 3 summarizes the quantization error results in terms of SF changes applied to synthetic and real datasets learning. As seen, FMIG algorithm presents the minimum of FQE compared to MIGSOM, GSOM, FKCN and fuzzy GSOM. Best results are reached for SF = 0.9. Indeed, NGT which is function of the spread factor decreases with the increase of SF. In addition, low values of NGT allow the maximum of huge data training to provide the appropriate map (well-spread). Besides, highlighted quantization error results are detected with real datasets (DNA, Letter and NE) where the decrease of FQE values is significant.

Table 3 Quantization error

SF	Dataset	FMIG	MIGSOM	GSOM	Fuzzy GSOM	FKCN
0.2	Dataset1	0.231	0.300	0.387	0.329	0.362
	Dataset2	0.205	0.293	0.413	0.353	0.333
	DNA	0.392	0.414	0.541	0.450	0.461
	Letter	0.446	0.483	0.654	0.506	0.499
	NE	0.201	0.290	0.395	0.309	0.290
0.4	Dataset1	0.174	0.215	0.280	0.228	0.222
	Dataset2	0.157	0.212	0.257	0.233	0.224
	DNA	0.290	0.386	0.505	0.419	0.400
	Letter	0.387	0.404	0.621	0.492	0.487
	NE	0.165	0.207	0.368	0.277	0.269
0.6	Dataset1	0.030	0.045	0.186	0.107	0.121
	Dataset2	0.033	0.049	0.205	0.155	0.112
	DNA	0.241	0.333	0.478	0.398	0.358
	Letter	0.339	0.346	0.501	0.474	0.490
	NE	0.124	0.132	0.299	0.255	0.243
0.8	Dataset1	0.024	0.030	0.094	0.089	0.030
	Dataset2	0.022	0.038	0.137	0.133	0.072
	DNA	0.239	0.307	0.381	0.370	0.315
	Letter	0.328	0.331	0.488	0.451	0.474
	NE	0.110	0.123	0.229	0.219	0.211
0.9	Dataset1	0.012	0.021	0.092	0.087	0.027
	Dataset2	0.017	0.025	0.136	0.132	0.071
	DNA	0.202	0.301	0.370	0.363	0.311
	Letter	0.301	0.302	0.471	0.445	0.472
	NE	0.075	0.095	0.220	0.219	0.206

It is noted that fuzzy GSOM algorithm in NE training reaches its best FQE value (0.219) at SF = 0.8. Afterwards, the increase of SF at 0.9 is without effect on fuzzy GSOM results (FQE = 0.219). This means that the growing process achieved the 500 iterations to train NE data regardless its stop condition (Eq. (**10**)) which depends on SF values. In other words, fuzzy GSOM growing does not reach the stop condition with 500 iterations; it needs more than such value to represent the data quantization. This situation is explained by the huge size aspect of NE that should have with such algorithm, more than 500 iterations to be correctly trained.

Table 4 shows that the topologic error is improved by the increase of SF which decreases the growing threshold NGT. More precisely, FTE with SF = 0.9, gives minimum values for Dataset1, Dataset2, DNA, Letter and NE. Moreover, FMIG results prove that this algorithm is capable of learning huge data and extracting their real topologic structure. Compared to the tested techniques, FMIG provides the best results in term of topologic error which presents the minimum values with FMIG learning.

Table 4 Topologic error

SF	Dataset	FMIG	MIGSOM	GSOM	Fuzzy GSOM	FKCN
0.2	Dataset1	0.266	0.305	0.402	0.335	0.332
	Dataset2	0.190	0.315	0.390	0.358	0.373
	DNA	0.205	0.265	0.400	0.392	0.386
	Letter	0.363	0.397	0.616	0.445	0.452
	NE	0.200	0.308	0.409	0.366	0.350
0.4	Dataset1	0.204	0.285	0.328	0.303	0.306
	Dataset2	0.157	0.270	0.319	0.299	0.367
	DNA	0.183	0.240	0.375	0.338	0.346
	Letter	0.302	0.339	0.489	0.416	0.412
	NE	0.168	0.232	0.371	0.251	0.284
0.6	Dataset1	0.150	0.194	0.285	0.215	0.240
	Dataset2	0.146	0.232	0.294	0.227	0.333
	DNA	0.188	0.217	0.322	0.369	0.356
	Letter	0.260	0.315	0.454	0.407	0.407
	NE	0.155	0.214	0.310	0.214	0.241
0.8	Dataset1	0.131	0.177	0.179	0.141	0.177
	Dataset2	0.121	0.165	0.230	0.206	0.305
	DNA	0.105	0.150	0.313	0.303	0.312
	Letter	0.228	0.244	0.426	0.400	0.402
	NE	0.142	0.207	0.283	0.198	0.209
0.9	Dataset1	0.124	0.135	0.136	0.130	0.161
	Dataset2	0.101	0.104	0.201	0.197	0.300
	DNA	0.095	0.144	0.300	0.291	0.304
	Letter	0.200	0.206	0.398	0.395	0.400
	NE	0.134	0.169	0.177	0.172	0.205

Dead unit percentages are shown in Table 5. As seen, the increase of the spread factor has a direct effect on the inactive units in the created map. From SF = 0.2 to SF = 0.9, we highlighted the decrease of dead unit percentages for all learning algorithms. These results prove that the trained datasets are correctly represented and visualized by their resulting maps. Besides, FMIG produces the minimum number of dead units compared to the rest of algorithms. We can explain such effectiveness by the fuzzy architecture of FMIG algorithm that is capable of representing real data. However, in the training of Dataset1 and Dataset2 all the algorithms produce the minimum of dead units at SF = 0.8 (no change for SF = 0.9). This stability is detected because the training of such datasets generates its final dead units at a number of iterations lower than 500. We note that the inactive unit percentage is without changes at a certain level of iterations.

Table 5 Dead unit percentage

SF	Dataset	FMIG	MIGSOM	GSOM	Fuzzy GSOM	FKCN
0.2	Dataset1	2.8	3.9	18.6	17.5	15.2
	Dataset2	1.6	3.2	12.3	11.7	10.8
	DNA	5.2	7.5	15.7	14	14.7
	Letter	5.7	9.8	14.3	13.8	13.3
	NE	12.9	16.4	21	18.4	17
0.4	Dataset1	1.3	3.7	14.4	13.1	12.9
	Dataset2	2.5	3	12	10.3	10.3
	DNA	5	7.2	14.8	13.5	13.2
	Letter	5.5	8.6	13.8	13	12.6
	NE	10.6	15.9	20.5	16.6	16.1
0.6	Dataset1	1.4	2.2	13.9	12.8	12.6
	Dataset2	1.3	2.5	11.6	10.1	9.9
	DNA	3.8	7.1	14	12.2	12
	Letter	4.9	9.5	12.5	11.7	11.8
	NE	8	14.8	18.8	15.9	15
0.8	Dataset1	0.9	1.8	12.2	11.3	11.2
	Dataset2	1	2.2	11	9.9	9.5
	DNA	3.2	6.6	13.5	12	11.5
	Letter	4.3	9	12.3	10.4	9.8
	NE	6.5	13.6	17.4	15.2	14.4
0.9	Dataset1	0.9	1.8	12.2	11.3	11.1
	Dataset2	1	2.2	10.9	9.8	9.5
	DNA	2.7	6.4	13.1	11.6	10.9
	Letter	4	8.7	11.8	9.9	9.6
	NE	5.9	12.9	17.1	15	14.2

5 Conclusion

Based on the spread factor changes, we have presented in this paper a special study in the control of the network growing. In fact, we have highlighted the effect of the spread factor on some approaches, FMIG, MIGSOM, GSOM, FKCN and fuzzy GSOM. We have contrasted this effect with the learning of huge datasets regarding topology, quantization and dead unit parameters. We proved that the network spread represented by the growing threshold is sensitive to the changes of SF. Indeed, the increase of SF ameliorates the preservation of data topology and quantization, and minimizes the dead units. FMIG algorithm provides the best results in comparison with the studied algorithms. It is important to point out that this study presents a significant survey for the selection of predefined parameters that involves a serious problem in huge data learning. Indeed, the spread factor has a direct effect on the growing of the oriented neural maps in the directions of input datasets to produce well distributed topographic maps with topology preservation.

Future research should focus on improving the spread control of the map over huge data learning by presenting new forms of growing threshold with the use of kernelized distance. It will be commode to incorporate evolutionary techniques in providing a new method of Fuzzy evolutionary SOM.

Acknowledgments. The authors would like to acknowledge the financial support of this work by grants from General Direction of Scientific Research (DGRST), Tunisia, under the ARUB program.

References

1. Tlili, M., Ayadi, T., Hamdani, T.M., Alimi, A.M.: Performance evaluation of FMIG clustering using fuzzy validity indexes. Soft Comput. **19**(12), 3515–3528 (2015)
2. Tlili, M., Hamdani, T.M., Alimi, A.M.: Big data clustering validity. In: SoCPaR, Tunisia, pp. 348–352 (2014)
3. Tlili, M., Hamdani, T.M., Alimi, A.M.: FMIG: fuzzy multilevel interior growing self-organizing maps. In: IEEE International Conference on Tools with Artificial Intelligence (ICTAI), Athens (2012)
4. Ayadi, T., Hamdani, T.M., Alimi, A.M.: A new data topology matching technique with multilevel interior growing self-organizing maps. In: IEEE International Conference on Systems, Man, and Cybernetics, pp. 2479–2486 (2010)
5. Ayadi, T., Hamdani, T.M., Alimi, A.M.: MIGSOM: multilevel interior growing self-organizing maps for high dimensional data clustering. Neural Process. Lett. **36**, 235–256 (2012)
6. Ayadi, T., Hamdani, T.M., Alimi, A.M., Khabou, M.A.: 2IBGSOM: interior and irregular boundaries growing self-organizing maps. In: IEEE sixth International Conference on Machine Learning and Applications, pp. 387–392 (2007)
7. Ayadi, T., Hamdani, T.M., Alimi, A.M.: On the use of cluster validity for evaluation of MIGSOM clustering. In: ISCIII: 5th International Symposium on Computational Intelligence and Intelligent Informatics, pp. 121–126 (2011)

8. Alimi, A.M.: Beta neuro-fuzzy systems. TASK Q. J. **7**(1), 23–41 (2003). Special Issue on Neural Networks edited by W. Duch and D. Rutkowska
9. Alimi, A.M., Hassine, R., Selmi, M.: Beta fuzzy logic systems: approximation properties in the MIMO case. Int. J. Appl. Math. Comput. Sci. **13**(2), 225–238 (2003)
10. Hamdani, T.M., Alimi, A.M., Khabou, M.A.: An iterative method for deciding SVM and single layer neural network structures. Neural Process. Lett. **33**(2), 171–186 (2011)
11. Hamdani, T.M., Alimi, A.M., Karray, F.: Enhancing the structure and parameters of the centers for BBF fuzzy neural network classifier construction based on data structure. In: Proceedings of IEEE International Join Conference on Neural Networks, IJCNN, Hong Kong, pp. 3174–3180 (2008). Art. no. 4634247
12. El Malek, J., Alimi, A.M., Tourki, R.: Problems in pattern classification in high dimensional spaces: behavior of a class of combined neuro-fuzzy classifiers. Fuzzy Sets Syst. **128**(1), 15–33 (2002)
13. Pascual, A., Barcena, M., Merelo, J.J., Carazo, J.M.: Mapping and fuzzy classification of macromolecular images using self-organizing neural networks. Ultramicroscopy **84**, 85–99 (2000)
14. Kohonen, T.: Self-organized formation of topologically correct feature maps. Biol. Cybern. **43**, 59–69 (1982)
15. Kohonen, T.: The self-organizing map. Neurocomputing **21**, 1–6 (1998)
16. Kohonen, T.: Self-organizing Maps. Springer, Berlin (2001)
17. Alahakoon, L.D., Halgamuge, S.K., Sirinivasan, B.: Dynamic self organizing maps with controlled growth for knowledge discovery. IEEE Trans. Neural Netw. **11**(3), 601–614 (2000). Special Issue on Knowledge Discovery and Data Mining
18. Fritzke, B.: Growing cell structure a self organizing network for supervised and un-supervised learning. Neural Netw. **7**, 1441–1460 (1994)
19. Amarasiri, R., Alahakoon, D., Smith, K.A.: HDGSOM: a modified growing self-organizing map for high dimensional data clustering. In: Fourth International Conference on Hybrid Intelligent Systems (HIS 2004), pp. 216–221 (2004)
20. Hsu, A.L., Halgarmuge, S.K.: Enhanced topology preservation of dynamic self-organising maps for data visualization. In: IFSA World Congress and 20th NAFIPS International Conference, vol.3, pp. 1786–1791 (2001)
21. Blake, C.L., Merz, C.J.: UCI repository of machine learning databases. University of California, Department of Information and Computer Science, Irvine, CA (1998). http://www.ics.uci.edu/mlearn/mlrepository.html
22. Theodoridis, Y.: Spatial datasets U an unofficial collection (1996). http://www.dias.cti.gr/ytheod/research/datasets/spatial.htm

Author Index

A. Abraham et al. (eds.), *Proceedings of the 16th International Conference on Hybrid Intelligent Systems (HIS 2016)*, Advances in Intelligent Systems and Computing 552,
DOI 10.1007/978-3-319-52941-7