Lecture Notes in Computer Science 5744

Commenced Publication in 1973
Founding and Former Series Editors:
Gerhard Goos, Juris Hartmanis, and Jan van Leeuwen

Jong-Hwan Kim Shuzhi Sam Ge Prahlad Vadakkepat
Norbert Jesse Abdullah Al Manum
Sadasivan Puthusserypady K Ulrich Rückert
Joaquin Sitte Ulf Witkowski Ryohei Nakatsu
Thomas Braunl Jacky Baltes John Anderson
Ching-Chang Wong Igor Verner David Ahlgren (Eds.)

Advances in Robotics

FIRA RoboWorld Congress 2009
Incheon, Korea, August 16-20, 2009
Proceedings

Springer

Volume Editors

Jong-Hwan Kim, jhkim@ee.kaist.ac.kr
Shuzhi Sam Ge, samge@nus.edu.sg
Prahlad Vadakkepat, prahlad@nus.edu.sg
Norbert Jesse, norbert.jesse@udo.edu
Abdullah Al Manum, eleaam@nus.edu.sg
Sadasivan Puthusserypady K, elespk@nus.edu.sg
Ulrich Rückert, ulrich.rueckert@hni.uni-paderborn.de
Joaquin Sitte, j.sitte@qut.edu.au
Ulf Witkowski, ulf.witkowski@hni.uni-paderborn.de
Ryohei Nakatsu, idmdir@nus.edu.sg
Thomas Braunl, tb@ee.uwa.edu.au
Jacky Baltes, jacky@cs.umanitoba.ca
John Anderson, andersj@cs.umanitoba.ca
Ching-Chang Wong, wong@ee.tku.edu.tw
Igor Verner, ttrigor@tx.technion.ac.il
David Ahlgren, david.ahlgren@trincoll.edu

Library of Congress Control Number: Applied for

CR Subject Classification (1998): I.2.9, H.5, J.4, K.4, K.3, K.8

LNCS Sublibrary: SL 6 – Image Processing, Computer Vision, Pattern Recognition, and Graphics

ISSN 0302-9743
ISBN-10 3-642-03982-0 Springer Berlin Heidelberg New York
ISBN-13 978-3-642-03982-9 Springer Berlin Heidelberg New York

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India
Printed on acid-free paper SPIN: 12741564 06/3180 5 4 3 2 1 0

Preface

This volume is an edition of the papers selected from the 12th FIRA RoboWorld Congress, held in Incheon, Korea, August 16–18, 2009.

The Federation of International Robosoccer Association (FIRA – www.fira.net) is a non-profit organization, which organizes robotic competitions and meetings around the globe annually. The RoboSoccer competitions started in 1996 and FIRA was established on June 5, 1997. The Robot Soccer competitions are aimed at promoting the spirit of science and technology to the younger generation. The congress is a forum in which to share ideas and future directions of technologies, and to enlarge the human networks in robotics area.

The objectives of the FIRA Cup and Congress are to explore the technical development and achievement in the field of robotics, and provide participants with a robot festival including technical presentations, robot soccer competitions and exhibits under the theme "Where Theory and Practice Meet."

Under the umbrella of the 12th FIRA RoboWorld Congress Incheon 2009, six international conferences were held for greater impact and scientific exchange:

- 6th International Conference on Computational Intelligence, Robotics and Autonomous Systems (CIRAS)
- 5th International Symposium on Autonomous Minirobots for Research and Edutainment (AMiRE)
- International Conference on Social Robotics (ICSR)
- International Conference on Advanced Humanoid Robotics Research (ICAHRR)
- International Conference on Entertainment Robotics (ICER)
- International Robotics Education Forum (IREF)

This volume consists of selected quality papers from the six conferences. The volume is intended to provide readers with the recent technical progresses in robotics, human–robot interactions, cooperative robotics and the related fields.

The volume has 31 papers from the 115 contributed papers at the FIRA RoboWorld Congress Incheon 2009. This volume is organized into seven sections:

- Emotion and Behavior, Human–Robot Interaction, Biped / Humanoid Robotics, Localization, Path Planning, Obstacle Avoidance, Control, Communication, Terrain mapping, and, Classification.

The editors hope that this volume is informative to the readers. We thank Springer for undertaking the publication of this volume.

Prahlad Vadakkepat

Organization

Honorary Chair

Jong Hwan Kim	KAIST, Korea

General Chair

Shuzhi Sam Ge	National University of Singapore, Singapore

Program Chair

Prahlad Vadakkepat	National University of Singapore, Singapore

Finance Chair

Hyun Myung	KAIST, Korea

Conferences Committee

CIRAS - International Conference on Computational Intelligence, Robotics and Autonomous Systems

General Chair

Prahlad Vadakkepat	National University of Singapore, Singapore

Program Chairs

Abdullah Al Manum	National University of Singapore, Singapore
Sadasivan Puthusserypady K	National University of Singapore, Singapore

AMiRE - International Symposium on Autonomous Minirobots for Research and Edutainment

General Chair

Ulrich Ruckert	University of Paderborn, Germany

Program Chairs

Joaquin Sitte	Queensland University of Technology, Austrailia
Ulf Witkowski	University of Paderborn, Germany

ICSR - International Conference on Social Robotics

General Chair

Ryohei Nakatsu National University of Singapore, Singapore

Co-general Chairs

Oussama Khatib Stanford University, USA
Hideki Hashimoto The University of Tokyo, Japan

Program Chair

Thomas Braunl The University of Western Australia, Australia

Program Committee Members

Arvin Agah, USA
Fuchun Sun, China
Robert Babuska, New Zealand
Jochen Triesch, Germany
Norbert Oswald, Germany
Dirk Wollherr, Germany
Rinhard Klette, New Zealand
Edwige Pissaloux, France
Gerhard Sagerer, Germany
Guilherme DeSouza, USA
Yaochu Jin, Germany
Masahiro Shiomi, Japan
Marcelo H. Ang, Sigapore
Maria Chiara Carrozza, Italy
Elizabeth Croft, Cananda
Kerstin Dautenhahn, UK
Kerstin Severinson-Eklundh, Sweden
Vanessa Evers, New Zealand
Guido Herrmann, UK
Sandra Hirche, Japan
Lars Hildebrand, Germany
Guy Hoffmann, USA
Frederic Maire, Australia
Carlo Alberto Avizzano
Guy Matko, Slovenia
Tom Ziemke, Sweden
Matthias Scheutz, USA
Hong Qiao, China
Nilanjan Sarkar, USA
Gil Weinberg, USA
Olivier Simonin, France
Frantisek Solc, Czech Republic
Chrystopher Nehaniv, UK
Lola Canamero, UK
John John Cabibihan, Singapore

ICAHRR - International Conference on Advanced Humanoid Robotics Research

General Chair

Jacky Baltes University of Manitoba, Canada

Program Chair

John Anderson University of Manitoba, Canada

ICER - International Conference on Entertainment Robotics

General Chair

Norbert Jesse Technische Universität Dortmund, Germany

Program Chair

Ching-Chang Wong Tamkang University, Taiwan

IREF - International Robotics Education

General Chair

Igor Verner Israel Institute of Technology, Israel

Program Chair

David Ahlgren Trinity College, USA

Table of Contents

Plenary Session

Emotion and Behaviour

Human Robot Interaction

Biped / Humanoid Robotics

Localization, Path Planning, Obstacle Avoidance

Control, Communication

Terrain Mapping

Classification

Intelligent Systems and Its Applications in Robotics

Okyay Kaynak

UNESCO Chair on Mechatronics, Boğaziçi Üniversitesi, Bebek, 34342 Istanbul, Turkey
okyay.kaynak@boun.edu.tr

Abstract. The last decade of the last millennium is characterized by what might be called the intelligent systems revolution, as a result of which, it is now possible to have man made systems that exhibit ability to reason, learn from experience and make rational decisions without human intervention. Prof. Zadeh has coined the word MIQ (machine intelligence quotient) to describe a measure of intelligence of man-made systems. In this perspective, an intelligent system can be defined as a system that has a high MIQ.

The presentation dwells upon how to define "intelligence" of man-made machines and continues with a discussion on computational intelligence, its main components and compares it with biological and artificial intelligence. The paradigm shift from computation with measurements to computation with perceptions is also pointed out to. The state-of-art reached in intelligent systems is overviewed with examples and a perspective on the future is given, based on "futurology" rather than "prophecy". The reasons behind the slow pace of developments are discussed, tying it to the availability of the computing power. The trends seen in this respect over the last century are overviewed and it is argued that the Moore's Law will have to reach an end, not so much because of technological difficulties but because of financial ones. Quantum and molecular computing are offered as possible alternatives. The talk closes with a consideration of the possible research directions in mechatronics and robotics as driving forces behind the development of intelligent systems.

J.-H. Kim et al. (Eds.): FIRA 2009, LNCS 5744, p. 1, 2009.

Neurodynamic Optimization and Its Applications in Robotics

Wang Jun

Mechanical and Automation Engineering, The Chinese University of Hong Kong,
Department of Mechanical and Automation Engineering,
The Chinese University of Hong Kong, Shatin, NT, Hong Kong
jwang@mae.cuhk.edu.hk

Abstract. Optimization problems arise in a wide variety of scientific and engineering applications. It is computationally challenging when optimization procedures have to be performed in real time to optimize the performance of dynamical systems. For such applications, classical optimization techniques may not be competent due to the problem dimensionality and stringent requirement on computational time. One very promising approach to dynamic optimization is to apply artificial neural networks.

Because of the inherent nature of parallel and distributed information processing in neural networks, the convergence rate of the solution process is not decreasing as the size of the problem increases. Neural networks can be implemented physically in designated hardware such as ASICs where optimization is carried out in a truly parallel and distributed manner. This feature is particularly desirable for dynamic optimization in decentralized decision-making situations arising frequently in robotics and control. In this talk, I will present the historic review and the state of the art of neurodynamic optimization models and selected applications in robotics. Specifically, starting from the motivation of neurodynamic optimization, we will review various recurrent neural network models for optimization. Theoretical results about the stability and optimality of the neurodynamic optimization models will be given along with illustrative examples and simulation results. It will be shown that many fundamental problems in robotics, such as robot motion planning and obstacle avoidance, can be readily solved by using the neurodynamic optimization models.

J.-H. Kim et al. (Eds.): FIRA 2009, LNCS 5744, p. 2, 2009.

Vision-Based Control of Robot Motion

Seth Hutchinson

Electrical and Computer Engineering, University of Illinois
Beckman Institute, University of Illinois, Urbana, IL 61801, USA
seth@uiuc.edu

Abstract. Visual servo control is now a mature method for controlling robots using real-time vision feedback. It can be considered as the fusion of computer vision, robotics and control, and it has been a distinct field since the 1990's, though the earliest work dates back to the 1980's. Over this period several major, and well understood, approaches have evolved and have been demonstrated in many laboratories around the world.

Many visual servo schemes can be classified as either position-based or image-based, depending on whether camera pose or image features are used in the control law. This lecture will review both position-based and image-based methods for visual servo control, presenting the basic derivations and concepts, and describing a few of the performance problems faced by each. Following this, a few recent and more advanced methods will be described. These approaches essentially partition the control system either along spatial or temporal dimensions. The former are commonly referred to as hybrid or partitioned control systems, while the latter are typically referred to as switched systems.

J.-H. Kim et al. (Eds.): FIRA 2009, LNCS 5744, p. 3, 2009.

Korea's Robotland: Merging Intelligent Robotics Strategic Policy, Business Development, and Fun

Eugene Jun

Incheon IT Industry Promotion Agency, 20F. Incheon IT Tower, 592-5, Dowha 1 Dong, Namgu, Incheon City, Korea, 402-711,
rightjun@iitpa.or.kr, sciencejun@yahoo.co.kr

Abstract. South Korea specializes in the design and manufacture of service and entertainment robots for consumer use. The government of South Korea considers robotics one of the countries main growth industries. To boost robot industry and accelerate the social demand of intelligent robots, a theme park titled 'Robotland' is being constructed in Korea near Incheon International airport, the gateway to Seoul. A total of $700 million will be invested by the Korean government, the city of Incheon and financial investors. The 760,000 square meter site (188 acres) will contains a number of displays featuring famous robots and robot characters from around the world. Ultra-modern rides and amusement facilities, exhibition halls, shopping arcades and hotels, featuring advanced ubiquitous environments, will also be included in the park. In addition, Robotland will contain R&D centers and educational institutions. The Grand opening is scheduled in 2012. In this session, attendees will hear of development plans for Robotland, along with the social and business issues that are driving the development of the world's first theme park.

Keywords: Lobotland, Robot, Teme park, Incheon, URC.

J.-H. Kim et al. (Eds.): FIRA 2009, LNCS 5744, p. 4, 2009.

Detecting Underlying Stance Adopted When Human Construe Behavior of Entities

Kazunori Terada, Kouhei Ono, and Akira Ito

Gifu University, 1-1, Yanagido, Gifu, 501-1193, Japan
terada@gifu-u.ac.jp

Abstract. Whether or not humans can construe the behaviors of entities depends on their psychological stance. The philosopher Dennett proposed human cognitive strategies (three stances) in which humans construe the behavior of other animated objects, including other humans, artifacts, and physical phenomena:'intentional', 'design' and 'physical' stances. Detecting the psychological stance taken toward entities is difficult, because such mental state attribution is a subjective cognitive process and hard to measure. In the present study, we proposed a novel method for detecting underlying stance adopted when human construe behavior of entities. In our method the subject was asked to select the most suitable action sequence shown in three movies each of which representing Dennett's three stances. To valid our method we have conducted an experiment in which the subjects were presented thirty short videos and asked to compare them to the three movies. The result indicated that the subjects did not focused on prior knowledge about the entity but could focused on motion characteristics per se, owing to simple and typical motion of an abstract shaped object.

1 Introduction

Behavior of an autonomous and high degree of freedom agent is ambiguous and it is sometimes misunderstood by those who observed it. Therefore, considering behavior of an agent which is easily and intuitively construed is important in human-agent interaction design. Utilizing psychological abilities of human used in human-human interaction is efficient to design such behavior. Many researchers suggest that gaze and gestures of robot facilitate human-robot communication. Utilizing the gaze and the gestures in human-agent interaction is based on the idea that using the metaphor implying human embodiment is efficient way to cause a human to treat an agent as a human. The Media equation [15] suggests that not only the appearance but also social cues, such as team identity and reciprocity, can cause a human to treat an agent (computer) as a human. Thus, anthropomorphising is a strong strategy for us to predict and understand other's behaviors.

However we sometimes adopt another strategies to predict and understand behavior of entities. According to Daniel Dennett [5], there are three different strategies that we might use when confronted with objects or systems including

J.-H. Kim et al. (Eds.): FIRA 2009, LNCS 5744, pp. 5–12, 2009.

other humans, artifacts, and physical phenomena: the physical stance, the design stance, and the intentional stance. Each of these strategies is predictive. We use them to predict and thereby to explain the behavior of the entity in question. The details of these three stances are described as follows.

Intentional stance. The intentional stance is closely related to the anthropomorphising. When adopting the intentional stance towards an entity, we attempt to explain and predict its behavior by treating it as if it were a rational agent whose actions are governed by its intentional mental states. The term "intentional stance" refers to our ability to attribute the full set of intentional states (beliefs, desires, thoughts, intentions, hopes, memories, fears, promises, etc.) not just to the specific mental state of intention [1]. Suppose you are sleeping in a bed and you feel someone shake your shoulder. You might understand his behavior in terms of intention "wake you up". You never care about the mechanism of arm movement or joint angle. Your main interest is what he want to do. "Attributing mental states to a complex system (such as a human being) is by far the easiest way of understanding it" ([1], p21).

Design stance. When we make a prediction from the design stance, we assume that the entity in question has been designed in a certain way, and we predict that the entity will thus behave as designed. Suppose you set your alarm clock for 7:00 a.m. in the evening. You predict that it will behave as designed: i.e., that it will buzz at 7:00 in the next morning. We are predicting the action of an alarm clock according to its design principle. You do not need to know anything about the physical constitution of the alarm clock in order to make this prediction. Furthermore, unlike intentional stance, you never think as it has an intention to wake you up. Design stance predictions are predictions based on what the system is designed to do.

Physical stance. The physical stance stems from the perspective of the physical sciences. To predict the behavior of a given entity according to the physical stance, we use information about its physical constitution in conjunction with information about the laws of physics. When you are sleeping in the bed and something fell down on you. You might understand the object's behavior in terms of Newtonian law. Like design stance, you never think as it has an intention to wake you up. However, it is very tedious to apply a physical stance to a complex artificial system like an alarm clock.

Detecting which of these three stances is adopted to entities is difficult because such mental state attribution is a subjective cognitive process and hard to measure. Although, much attention is paid to detecting whether human takes intentional stance or not and what cues trigger intention attribution in terms of Theory of Mind, there are few studies addressing the design and physical stance detection. Methods for detecting intentional stance is not well established. Self-reporting questionnaires are usually used to detect intention attribution. Observers report their subjective percepts in response to the animations[18][6][17]. Retrieval methods for subjective mental states, which rely on subjective self-reporting have been criticized for methodological limitations [10] and for that

higher-order cognitive processing is engaged in order to produce these descriptions [16]. Neuroimaging techniques such as functional magnetic resonance imaging (fMRI) and positron emission tomography (PET) seem to be useful for examining mental processes without relying on self-reporting. These methods, however, restrict subject's natural physical interactions due to the spatial and physical constraints of apparatus used in fMRI and PET. Furthermore, the area of the brain that activates intention attribution is unclear. Thus, so far, the measuring method for detecting subjective mental states about intention attribution has not been established.

In our previous study, we employed a questionnaire method for assessing whether human could attribute intention toward non-humanoid robot or not. Subjects were asked which of Dennett's three stances they took toward the robot, giving them a brief example corresponding to each stance. In our questionnaire method, there are same problems as described above. Subjects in our experiment asked to select the most suitable description which can represent the stance taken toward the robot by them during the experiment. This process required subjects to compare notions constructed by reading the descriptions and recalled stance. It is possible that subjects might have answered a typical stance according to a metacognitive view.

In this paper, to cope with this problem, we propose a novel method for detecting stance. In our method a subject was asked to select the most suitable action sequence shown in three animations each of which representing Dennett's three stances. To valid our method we have conducted an experiment in which subjects were presented thirty short videos and asked to compare them to the three animations.

2 Experiment

2.1 Video Stimulus

Table 1 shows the list of videos used in our experiment. The duration of the videos were at most 18 seconds. The entities acted in the videos are selected form the following six categories: human, animal, plant, simple machine and vehicle, complicated machine, and simple and natural object. A large variety of stimuli which invoke intention attribution are reported, including rationality [8], goal or goal-directedness [12], self-propelled motion [2][14][9], equifinality [4], spatial contingencies [3], and violations of Newtonian laws [11]. There are, however, few researches addressing the cues for taking design and physical stances. It seems apparent that when the motion of entities are governed by cirtain design principles or physical law. So, the motions of the entities in the videos presented to subjects were selected based on the following four criteria: a) it is self-propelled or not, b) it has sensors or not, c) it is autonomous or not, and d) it is goal-directed or not. None of all the simple and natural objects are self-propelled, having sensors, and autonomous. The goal-directedness of simple and natural objects is defined wheter or not the motion stops at a certain position or a state.

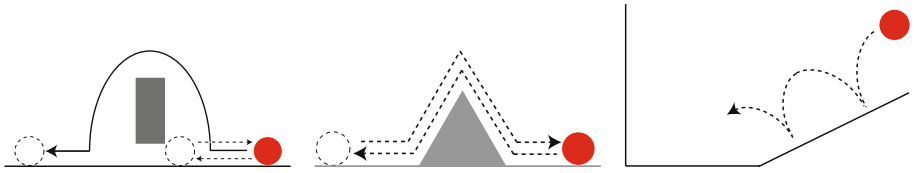
Table 1. Video stimulus used in our experiment

Entity	Goal-directedness	Motion
Human	○	1. A man putting on a jacket
		2. A woman picking up a ball
	-	3. A man going down a stairs
		4. A woman looking around
Animal	○	5. A cat eating food
		6. A dog playing with a balloon
	-	7. A yawning hamster
		8. A sleeping dog
Plant	○	9. A coiling tendril
	-	10. A blooming flower
		11. Pampas grasses blowing in wind
Complex machine	○	12. A humanoid robot solving a rubic cube
		13. A robotic arm assembling something
		14. A line tracing robot
	-	15. A Furby changing facial expression
		16. A crossing gate arm closing
		17. A robot raising and lowering both hands slowly
Simple machine	○	18. A clockwork-driven toy bear walking toward a stuffed bear
	-	19. A biped robot walking
		20. A oscillating electric fan
Vehicle	○	21. A cabin of an aerial tramway approaching a station
		22. A bus arriving at a station
	-	23. A running train
Simple object	○	24. A rolling empty can
		25. A rolling ball stops moving
	-	26. An empty can rolling down a slope
		27. A shishi-odoshi, a bamboo tube which is slowly filled with water and then emptied out, clacking against a stone
Natural object	○	28. A sheet of paper burning out
	-	29. A cloud floating in the sky
		30. Water falling down (waterfall)

The videos of a coiling tendril and a blooming flower were fast forwarded so that subjects recognize these motions.

2.2 Animated Action Sequences for Assessing Stances

Subjects were asked to chose an alternative from the three different action sequences of an circle, which is most suitable for representing the motion of the entity in the video. The details of these three animations are described as follows.



(a) Goal-directed action (intentional stance) (b) Mechanical motion (design stance) (c) Physical law (physical stance)

Fig. 1. Three motion patterns of a circle representing Dennett's three stances

Goal-directed action (intentional stance). In the animation corresponding to the intentional stance, a circle jumps over a rectangular obstacle, representing an intention of going to the other side of it (see Fig. 1(a)). Before the circle jumped over the rectangle, it tried to go under the rectangle because there is a gap between the ground and the rectancle. However the circle failed to do so because the gap is smaller than a diameter of the circle. Then the circle go back to the starting position and tried again. After the second fail of going under the lectangle, the circle jumped over it. The movement of the circle contains an intention in addition to having properties of goal-directedness[7][12], self-propulsion[13], rationality[7], and violation of Newtonian law[16] .

Mechanical motion (design stance). The cirlce in the animation corresponding to the design stance shows a reciprocating motion on the triangular shaped object (see Fig. 1(b)). Although the motion of the circle is self-propelled, rational, and goal-directed, they does not give strong impressions. The reciprocating motion is intended to give rise to an impression of being 'designed' to repeat mechanical motion.

Physical law (physical stance). The cirlce in the animation corresponding to the physical stance bounces down a slope, then bounces against a wall, decreased bouncing height, and finally stops (see Fig. 1(c)). The motion of the circle represents it is governed by the Newtonian law of gravitation.

2.3 Experimental Procedure

The questionnaire was described in HTML. A page contains a stimulus video and three alternatives of animation each of which represents one of three stance. Subjects were asked to select the most suitable action sequence for representing the motion of the entity in the video. Thirty pages each of which contains one video and three animations were presented to a subject in a random order. A Video was started when the page was exposed to a subject and repeated until he/she proceeded to the next page.

Thirty naive university students participated in the experiment. Subject ages ranged from 19 to 25 years.

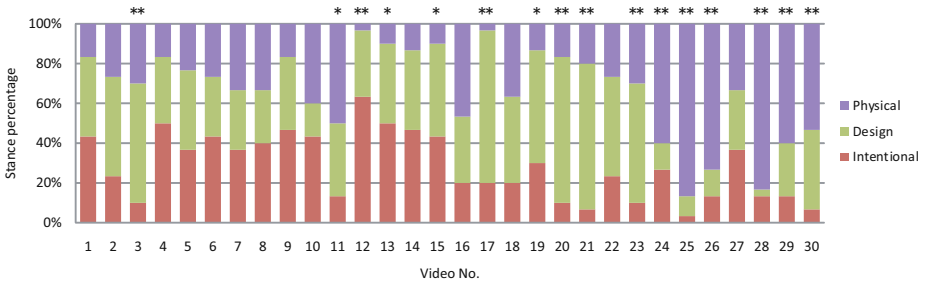


Fig. 2. Distributions of subjects' stances toward thirty videos

2.4 Experimental Result

Figure 2 shows distributions of subjects' stances toward thirty videos. Chi-square goodness-of-fit test revealed that the distributions of subjects' stances toward sixteen out of thirty videos were significantly different from equal ratio, 1:1:1.

3 Discussions

Subjects did not always took intentional stance toward human and animals. None of the motions performed by a human or an animal (video no. 1 to 8) was construed as goal-directed (intentional) by most subjects. At most 50% of subjects construed the video no. 4 (a woman looking around) as goal-directed. Moreover, the video no. 3 (a man goig down a stairs) construed as mechanical (design stance) by 60% of subjects. The difference of the two motions is that while in the video no. 3 a man just went down a stairs and there was no obvious goal, in the video no. 4 the action of looking around could give subjects an impression of having a will to look for something. The actions of putting on a jacket (no. 1) and picking up a ball (no. 2) were categorized into goal-directed actions by us. Many subjects, however, did not construed these actions as goal-directed. This may be because from these actions a subject did not perceive strong intentions.

It seems that people normally take intentional stance toward human and animals. The result indicated, however, that subjects did not always took intentional stance toward them. One of the reason is that our definition of intentional stance represented by the circle's goal-directed action may not suitable for assessing whether or not a subject takes intentional stance. Another reason is that all of the actions taken by the humans and animals did not have obvious goal, so subjects could not infer a particular intention from these actions.

The participants sometimes construed actions performed by robots as intentional. The video the most participants construed as intentional was no. 12 (a humanoid robot solving a rubic cube). Moreover, the most stance taken toward the video no. 13 (a robotic arm assembling something) and 14 (a line tracing

robot) by subjects was intentional stance. In these videos, there were more obvious goals pursued by the robots, such as solving a rubic cube or assembling something than the videos of human and animal.

The motions in the video no. 17, 20, and 21 were construed as mechanical. This is plausible because the entities in these video, a robot, an electric fan, and a cabin of an aerial tramway, are usually construed as designed object. Another explanation of this result is that the subjects might focused on the aspect of reciprocation because the robot in the video no. 17 raised and lowerd its hands, the fan head in the video no. 20 oscillated, and a cabin of an aerial tramway usually reciprocate between terminals even though the cabin in the video just arrived at the terminal. However, the motions in the video no. 19, 22, and 23 were not reciprocating but the most stance taken toward them was design stance. This means that the subjects did not focused on the aspect of reciprocation but mechanical motion.

The motions in the video no. 24, 25, 26, 28, 29, and 30 were construed as physical. Those who selected the animation representing physical stance might focused on Newton's law because the entities in the video no. 24, 25, 26, and 30 fell down according to gravity. However, flames in the video 28 and the cloud in the video 29 was not driven by gravity. So, the subjects might find a similarity in terms of that the motions are governed by general physical law.

Although the motion of the bamboo tube in the video 27 also governed by Newton's law, about 67% of subjects did not select the animation representing physical stance. This is because that the motion of the bamboo tube seems to be "designed to clack periodically" or "intended to act such a way".

4 Conclusions

In this paper, we proposed a novel method for detecting underlying stance adopted when human construe behavior of entities. In our method subjects were asked to select the most suitable action sequence shown in three animations each of which representing Dennett's three stances. To valid our method we have conducted an experiment in which subjects were presented thirty short videos and asked to compare them to the three animations. The result indicated that subjects did not forcused on prior knowledge about the entity but could forcused on motion characteristics per se, owing to simple and typical motion of an abstract shaped object.

References

1. Baron-Cohen, S.: Mindblindness: An Essay on Autism and Theory of Mind. The MIT Press, Cambridge (1995)
2. Baron-Cohen, S.: Mindblindness: An Essay on Autism and Theory of Mind. The MIT Press, Cambridge (1995)
3. Bassili, J.N.: Temporal and spatial contingencies in the perception of social events. Journal of Personality and Social Psychology 33(6), 680–685 (1976)

4. Csibra, G., Gergely, G., Bíró, S., Koós, O., Brockbank, M.: Goal attribution without agency cues: the perception of 'pure reason' in infancy. Cognition 72(3), 237–267 (1999)
5. Dennett, D.C.: The Intentional Stance. Bradford Books/MIT Press, Cambridge/Mass (1987)
6. Dittrich, W.H., Lea, S.E.G.: Visual perception of intentional motion. Perception 23(3), 253–268 (1994)
7. Gergely, G., Csibra, G.: Teleological reasoning in infancy: the naïve theory of rational action. Trends in Cognitive Science 7(7), 287–292 (2003)
8. Gergely, G., Nádasdy, Z., Csibra, G., Bíró, S.: Taking the intentional stance at 12 months of age. Cognition 56(2), 165–193 (1995)
9. Heider, F., Simmel, M.: An experimental study of apparent behavior. The American Journal of Psychology 57(2), 243–259 (1944)
10. Mar, R.A., Macrae, C.N.: Triggering the intentional stance. In: Novartis Foundation Symposium, vol. 278, pp. 111–132 (2007)
11. Opfer, J.E.: Identifying living and sentient kinds from dynamic information: the case of goal-directed versus aimless autonomous movement in conceptual change. Cognition 86(2), 97–122 (2002)
12. Premack, D., Premack, A.J.: Motor competence as integral to attribution of goal. Cognition 63(2), 235–242 (1997)
13. Premack, D.: The infant's theory of self-propelled objects. Cognition 36(1), 1–16 (1990)
14. Premack, D., Premack, A.J.: Moral belief: Form versus content. In: Mapping the mind: Domain specificity in cognition and culture, pp. 149–168. Cambridge University Press, Cambridge (1994)
15. Reeves, B., Nass, C.: The Media Equation: How People Treat Computers, Television, and New Media Like Real People and Places. CSLI Publications, Stanford (1998)
16. Scholl, B.J., Tremoulet, P.D.: Perceptual causality and animacy. Trends in Cognitive Science 4(8), 299–309 (2000)
17. Tremoulet, P.D., Feldman, J.: Perception of animacy from the motion of a single object. Perception 29(8), 943–951 (2000)
18. Tremoulet, P.D., Feldman, J.: The influence of spatial context and the role of intentionality in the interpretation of animacy from motion. Perception & Psychophysics 68(6), 1047–1058 (2006)

Motion and Emotional Behavior Design for Pet Robot Dog

Chi-Tai Cheng, Yu-Ting Yang, Shih-Heng Miao, and Ching-Chang Wong

Department of Electrical Engineering, Tamkang University
151, Ying-Chuan Rd. Tamsui, Taipei County, Taiwan 25137, R.O.C.
wong@ee.tku.edu.tw

Abstract. A pet robot dog with two ears, one mouth, one facial expression plane, and one vision system is designed and implemented so that it can do some emotional behaviors. Three processors (Inter® Pentium® M 1.0 GHz, an 8-bit processer 8051, and embedded soft-core processer NIOS) are used to control the robot. One camera, one power detector, four touch sensors, and one temperature detector are used to obtain the information of the environment. The designed robot with 20 DOF (degrees of freedom) is able to accomplish the walking motion. A behavior system is built on the implemented pet robot so that it is able to choose a suitable behavior for different environmental situation. From the practical test, we can see that the implemented pet robot dog can do some emotional interaction with the human.

Keywords: Robot Dog, Emotional Behavior.

1 Introduction

In the past, pet robots with lovely behaviors let them enter human's life [1]. The pixie "Furdy" on the market makes pet robots become more popular. But its processer is slower at that time. Until the recent year, the pet robot with more intelligence is beginning to be presented to the public [2-5]. The Sony AIBO [6,7] is the most advanced one in this area of robot dog.

A pet robot dog with 20 DOF (degrees of freedom) is designed and implemented so that it can do some emotional behaviors. In order to let the implemented pet robot have a high ability of environmental detection, a vision system (one USB camera), three touch sensor, and a temperature sensor are equipped on the body of the implemented robot to obtain the information of environment to decide an appropriate action. Three processors (Inter® Pentium® M 1.0 GHz, 8051 and NIOS) are used to control the robot. Inter processor is used to receive the vision from USB camera and to be a high -level controller to process the high level artificial intelligent. 8051 is an 8-bit processer used for some facial expression. NIOS is a 32-bit embedded soft-core processor used for the robot locomotion control. NIOS processor is used to be a low-level controller to control the walking and other actions. A control board with a FPGA chip and a 64 Mb flash memory are mainly utilized to control the robot. Many functions are implemented on this FPGA chip so that the design circuit is simple.

J.-H. Kim et al. (Eds.): FIRA 2009, LNCS 5744, pp. 13–22, 2009.

2 Mechanical Structure Design

The main design concept of a pet robot is to let its weight and size be light and compact, respectively. A pet robot dog with 20 DOF (degrees of freedom) is described in this section. The frameworks of pet robot are mainly fabricated from aluminum alloy 5052 in order to realize the concepts of light weight, wear-resisting, high stiffness, and wide movable range. Each actuator system for the joint consists of a high torque and a gear.

Mechanical structure design is the first step in the pet robot dog design. There are 20 degrees of freedom implemented by using the inertial coordinate system fixed on the ground. There are four rotational direction of joints are defined by using the passive joints and other 16 degrees of freedom are defined by using the active joint. There are 2 degrees of freedom on the neck, 2 degrees of freedom on the ear, and 4 degrees of freedom of each leg. The photograph of pet robot dog is shown in Fig. 1. The details of the development of the head and legs are described as follows:

Fig. 1. Photograph of the implemented pet robot dog

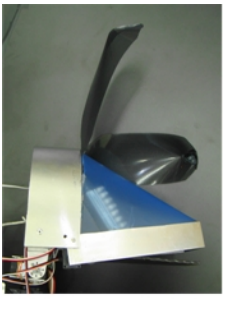


Fig. 2. Head design of the pet robot dog

In the head design, there are four degrees of freedom on it. Two degrees of freedom is adopted on the neck so that the head of the robot can turn right-and-left and up-and-down. Another two degrees of freedom is adopted on the ear. The picture of head design is described in Fig. 2. A LED dot-matrix with 8×16 LEDs as shown in Fig. 3 is design and implemented in the center of the head so that the robot can do some emotional interaction.

Fig. 3. The LED dot-matrix entity

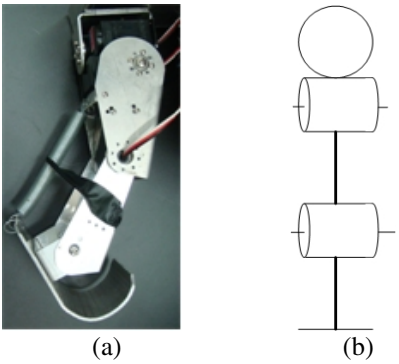


Fig. 4. Leg design of the pet robot dog

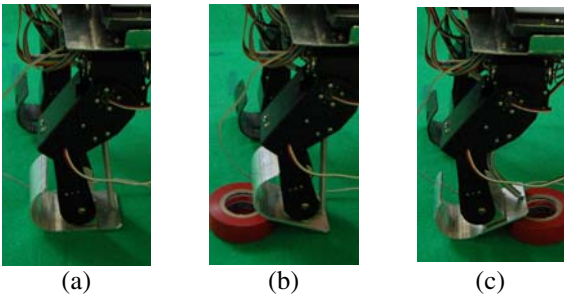


Fig. 5. Three states of the robot walks on an uneven floor: (a) general floor, (b) front rise, and (c) back rise

In the leg design, there are 16 degrees of freedom on the four legs. Each leg has 3 active degrees of freedom which are driven by servomotor. Another one degrees of freedom is driven by nothing which is moving passive. The picture of leg is described in Fig. 4. The passive joint is used for making the end of the leg always parallel to the floor. Two states of the robot walking on the uneven floor are shown in Fig. 5. We can see that the passive joint makes the robot more stable.

3 Electronic Structure Design

In the electronic design of the robot, the process device and the system block diagram are described in Fig. 6 and Fig. 7, respectively. In order to build a fully autonomous vision-based pet robot dog, a notebook is chosen to process the vision image of environment. The image of the field is captured by the camera.

Three touch sensors, one power detector, and one temperature detector sensor are mounted on the pet robot dog to obtain the touch state, power state, and temperature state of the robot, respectively. The touch sensor is mounted on the head to detect the touch behavior from the human. The power detector is mounted on the body to detect the power of the battery. The temperature detector sensor is mounted on the joint in the front leg to detect the temperature of the servomotor. These three kinds of detectors are used to present the states of the robot by some quantitative data. By analyzing these data, the robot emotion can be simulated. The relative signal is caught by a System On a Programmable Chip (SOPC), which is a FPGA chip and a 32-bit embedded soft core processor named NIOS in it is used to process the data and control the robot.

The circuit block diagram is described in Fig. 8. It can be reduced to three parts which are the LED dot-matrix circuit, the power detector circuit, and the temperature detector circuit. The LED dot-matrix circuit is designed by 128 LED, and controlled by an 8051 processer. It can express robot's feelings. Six kinds of feeling are designed

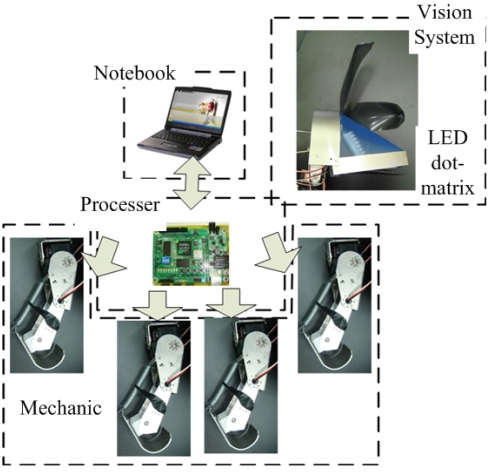


Fig. 6. The device of pet robot dog

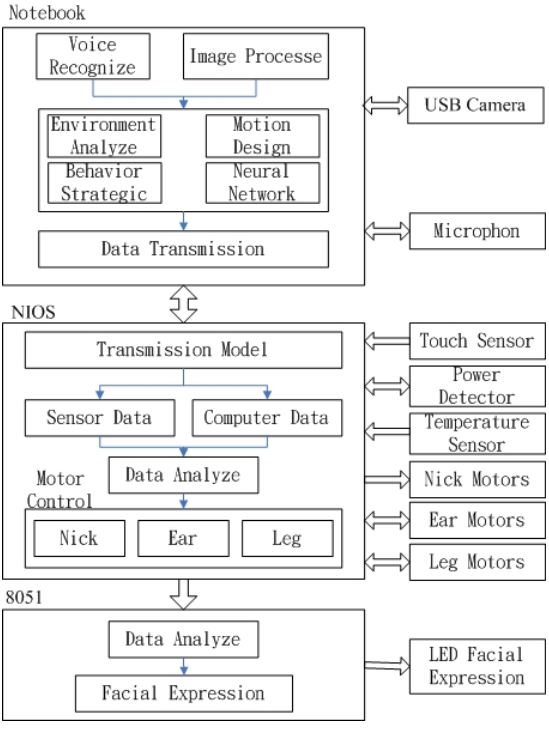


Fig. 7. System block diagram

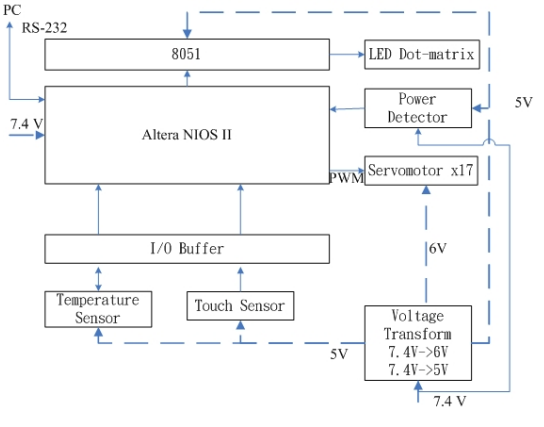


Fig. 8. Circuit block diagram

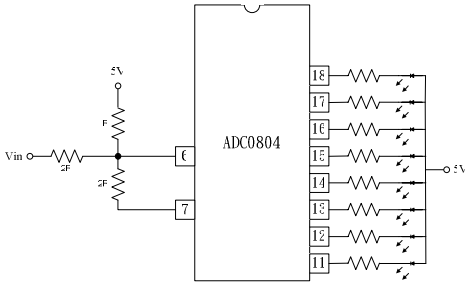


Fig. 9. Power detector circuit

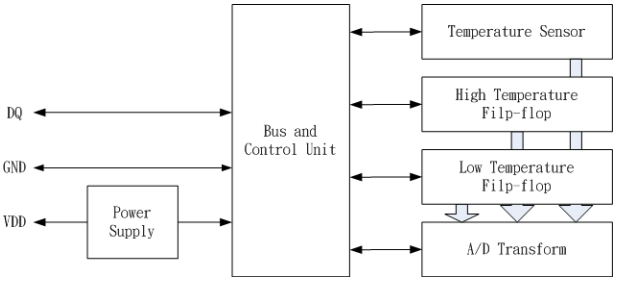


Fig. 10. Temperature detector circuit

in the LED dot-matrix. The power detector circuit is shown in Fig. 9, where ADC0804 is used to change analogy signal to an 8-bit digital signal. The temperature detector circuit is shown in Fig. 10. Sensor DS1821 is used to detect temperature, and the output is digital data.

4 Motion and Emotion Structure Design

A mechanical structure with 20 degrees of freedom is proposed to implement a robot dog. Two move motions including “step forward” and “turn right/left”, six non-move motions with ear, and six emotional expressions are designed for the robot’s movements. In order to verify the effectiveness of the implemented pet robot dog, three kinds of behaviors: Move Motions, Non-move Motions, and Emotional Expressions are carried out on a horizontal even plan and described as follows:

(a) Move Motions

Move motion is design to let the robot move. The implemented pet robot dog can do two kinds of motions: “step forward” and “turn right/left”. Some snapshots of the step forward walking of the pet robot are shown in Fig. 11.

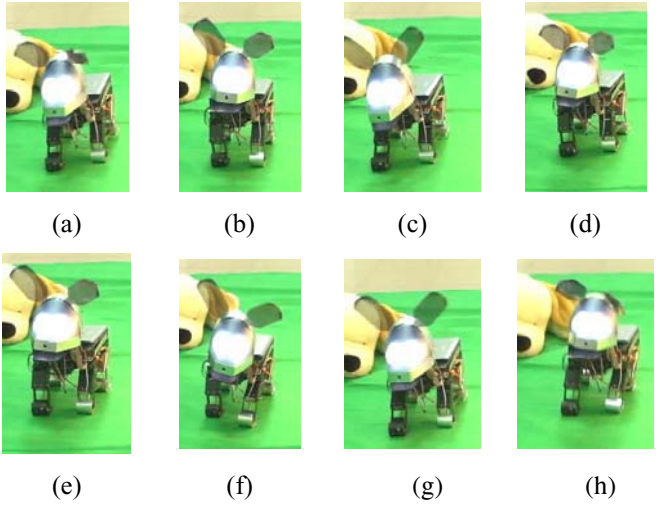


Fig. 11. Step forward motion

(b) Non-move Motions

Non-move motions are design to interact with the human. They are often used with the Emotional Expressions. Some snapshots of stretch motion and ear motion are shown in Fig. 12 and Fig. 13, respectively.

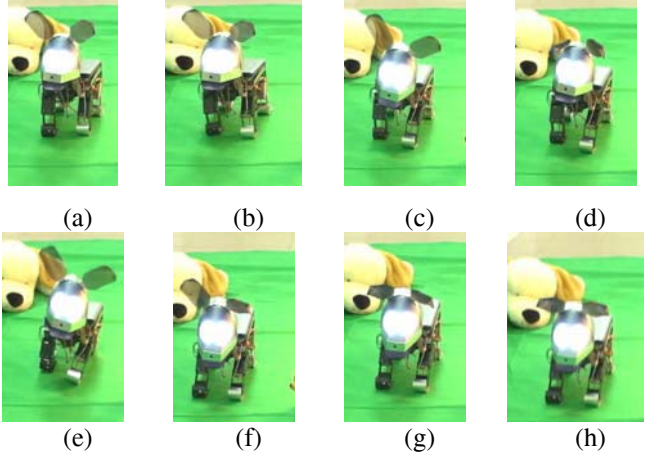


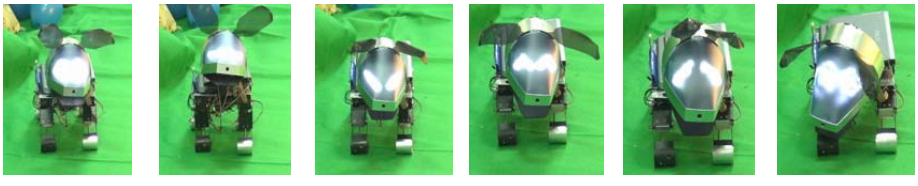
Fig. 12. Stretch motion

(a) (b) (c)

Fig. 13. Ear motion

(c) Emotional Expressions

Emotional Expressions is design to expression the emotion more direct. It makes the pet robot friendly. Some snapshots of Emotional Expressions are shown in Fig. 14.



(a) ∗ ∗ (b) v v (c) \ / (d) Λ Λ (e) / \ (f) ⊙⊙

Fig. 14. Emotional Expression

5 Behavior Design

There are four parts in the behavior design: (1) External Environment System, (2) Physiology System, (3) Psychology System, and (4) Behavior System. They are described as follows:

(1) External Environment System

External Environment System includes “touch sensor”, “vision system”, and “voice detector”. The main idea of External Environment System is the natural interface between human and robot. The natural interface makes the robot more like a real pet.

(2) Physiology System

Physiology System includes “seeing people”, “being touched”, “power quantity”, and “motor’s temperature”. The main idea of Physiology System is the body state of the robot. These quantitative data can let the robot understand its body state.

(3) Psychology System

Psychology System includes “hungry quantity”, “curious quantity”, “tire quantity”, and “vitality”. The main idea of Psychology System is to simulate the real dog feeling. Psychology System is affected by the Physiology System. The “hungry quantity” is affected by “power quantity”, the “curious quantity” is affected by “be

touched", and the "tire quantity" is affected by "motor's temperature". In order to design a realistic pet robot dog, "vitality" is designed to affect the "curious quantity" which makes robot personality.

(4) Behavior System

Behavior System includes "eat", "play intense", "play gently", and "sleep". The main idea of Behavior System is motion and response. Because of the "vitality" may be different, so the behavior may not always the same. The robot with higher "vitality" will play more with human. Reversely, the robot with lower "vitality" will appear emotionless.

Two experiment results of the robot in the high "tire quantity" state and the "sleep" state are shown in Fig.15. We can see that pet robot dog can do different emotion.

(a) (b) (c)

Fig. 15. The experiment results of the robot in the high "tire quantity" and the "sleep" state

6 Conclusions

A mechanical structure with 20 DOF is proposed so that the implemented pet robot dog can do some basic motion experiments. Furthermore, some emotional behaviors are design for it. In the future, a fuzzy system will be considered to let the pet robot dog have the behavior learning ability so that it is able to choose a better behavior for different environmental situation.

Acknowledgement

This research was supported in part by the National Science Council (NSC) of the Republic of China under contract NSC 97-2218-E-032-001.

References

1. Denavit, J., Hartenberg, R.S.: A kinematic notation for lower-pair mechanisms based on matrices. Transactions of ASME, Journal of Applied Mechanics 22, 215–221 (1955)
2. Paul, R.P.: Robot Manipulators: Mathematics Programming and Control. MIT Press, Cambridge (1981)
3. Hiros, S.: A study of design and control of a quadruped walking vehicle. International Journal of Robotics Research 3 (1984)
4. Blumberg, B.M., Galyean, T.A.: Multi-level direction of autonomous creatures for real-time virtual environments. In: Proceedings of ACM SIGGRAPH (1995)

5. Blumberg, B.M., Todd, P.T., Maes, P.: No Bad Dogs: Ethological Lessons for Learning in Hamsterdam. In: Proceedings of International Conference on Simulated Adaptive Behavior, pp. 295–304 (1996)
6. Kubota, N., Kojima, F., Fukuda, T.: Self-consciousness and Emotion for a Pet Robot with Structured Intelligence. In: Proceedings of IEEE IFSA World Congress, pp. 1786–2791 (2001)
7. Terzopoulos, D., Tu, X., Grzeszczuk, R.: Artificial fishes: Autonomous locomotion, perception, behavior, and learning in a simulated physical world. Journal of Artificial Life 1(4), 327–351 (1994)

Grounding Robot Autonomy in Emotion and Self-awareness

Ricardo Sanz, Carlos Hernández, Adolfo Hernando,
Jaime Gómez, and Julita Bermejo

Autonomous Systems Laboratory
Universidad Politécnica de Madrid, Spain
Ricardo.Sanz@upm.es
http://www.aslab.org

Abstract. Much is being done in an attempt to transfer emotional mechanisms from reverse-engineered biology into social robots. There are two basic approaches: the imitative display of emotion —e.g. to intend more human-like robots— and the provision of architectures with intrinsic emotion —in the hope of enhancing behavioral aspects. This paper focuses on the second approach, describing a core vision regarding the integration of cognitive, emotional and autonomic aspects in social robot systems. This vision has evolved as a result of the efforts in consolidating the models extracted from rat emotion research and their implementation in technical use cases based on a general systemic analysis in the framework of the ICEA and C3 projects. The desire for generality of the approach intends obtaining universal theories of integrated —autonomic, emotional, cognitive— behavior. The proposed conceptualizations and architectural principles are then captured in a theoretical framework: ASys — The Autonomous Systems Framework.

1 Introduction

Emotion is a hot topic these days; not only in psychology and neuroscience but in robotics and specially in social robotics. It has become a fashionable topic and a buzzword of effectiveness in getting funding. Roboticists look at the expression of emotion in humans or at the biology of emotion in animals in search for inspiration to build better performing systems.

The question of what "performing" means in this context is of extreme relevance, however. The anthropomorphizing shallow conception of emotion is quite dangerous because it leads to research done in robotization of emotions that is strictly limited to the generation of facial expression instead of focusing on core issues [1].

While somewhat valuable, this face-centric approach is reduced in most cases to just a set of useless construals of naïve conceptions of emotion instead of being done in terms of absolutely measurable system performance. Only from the perspective of human experience of the interaction with robots this approach

J.-H. Kim et al. (Eds.): FIRA 2009, LNCS 5744, pp. 23–43, 2009.

may have some sense, but in this case performance shall be measured using psychological tests in humans [2]. As an example of the class of objectives pursued in this type of research let's quote Breazeal [3]:

> In dramatic contrast, social (or sociable) robots are designed to engage people in an interpersonal manner, often as partners, in order to achieve social or emotional goals.

There is a big difference between someone having the opinion or the enjoyment that a robot displays human-like emotions and the existence of a scientific and technological substrate of them that can outperform non-emotional systems. Building robots that just display emotions —no matter however complex— may be not the proper way to build the long-term technological assets necessary to produce measurable increases of performance in social robots.

Analyzing and building emotions and indeed two very different activities. The endeavors of science and technology are quite different indeed and what is the product of one —theories— is part of the input materials for the processes of the second. Technology needs solid theories to base their designs upon and in the case of artificial emotion for social behavior these solid theories are lacking.

The long-term Autonomous Systems Project (ASys) intends not only the technology for robust autonomous behavior but also the necessary scientific conceptualizations to ground the technological implementations. This paper describes the theoretical positioning underpinning our research on integrated cognition-emotion architectures for the construction of versatile, adaptable, autonomous social robots.

2 Toward a Solid Ground for Autonomy and Socialization

The lack of a unified theory of emotion in science has produced an explosion of artificial intelligence and robotic models that have heterogeneously addressed the role of emotion in social systems behavior. These models have been expressed explicitly or implicitly —because any implementation is a reification of a theory, whether explicit or not.

Two main drives guide this activity of enacting emotions: trying to do science, *i.e.* improve our understanding of the world; and trying to do technology, *i.e.* improve our capability to change the world. In the case of social robots this last one may be the central focus, but the science supporting it is also necessary to transcend the craft phase of the technology.

Regarding the technological aspects, there are two basic approaches to the use of emotion in robotics: the *imitative display of emotion* —to intend more human-like robots— and the provision of *architectures with intrinsic emotion* —in the hope of enhancing behavioral aspects that go beyond the display of emotion, e.g. escape behaviors in presence of risk or approach behavior in the presence of potential collaborators [1]. This last one is what can produce qualitative changes in robot behavior and is the focus of this work.

The research described in this paper is based on the hypothesis that the architectures for enhanced social behavior cannot be different for the architectures for individual autonomy. Agents must be able to perform well in social and non-social contexts using the same architecture. This implies that the substrates and manifestations of emotion shall happen as a result of putting the autonomous agent in a social context and not by means of ad-hoc socialization emotional mechanics.

The ASys architectural approach for autonomous, self-aware robotics is based on the leveraging of a unified model of emotion as meta-control that scales up to the level of awareness of self and others (see Figure 1). This permits the leveraging of such intrinsic control mechanisms in both social- and non-social contexts.

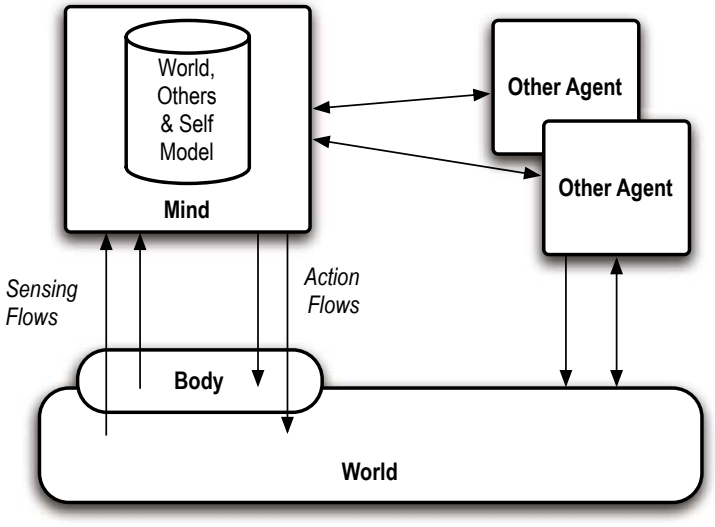


Fig. 1. The ASys architectural approach to social behavior is based on model-based —world, self and others— cognitive mechanisms for emotion and self-awareness

3 Methodological Positioning

From a scientific perspective, the enaction of models of emotion in social contexts is guided by the idea that if the proposed models generate artificial behavior that is similar to the behavior of a biological system —in the conditions that elicit some kind of emotion in it— the implemented models are good models of natural emotion.

This does not hold in general for several reasons, the three most important being:

1. The same behavior —*i.e.* input-output temporal correlation of quantities— can be generated by different dynamical structures. So identical behavior does not necessarily imply identical mechanics.
2. The conditions are not the same for an animal and a simulated animal in terms of *inputs* —because the perceptual systems and underlying ontologies are not assimilable— nor *outputs* —the environment that affects the two classes of agents doesn't change the same way.
3. The architectures are not the same for an animal and a robot in terms of the core architecture upon which emotion operates —while the environments can be identical.

This does not mean that simulation or robotization of models are useless for testing models; it means that we must take care to *discount the differences* and avoid non-sound abductions. It is necessary to get rid of the extended ad-hocery in computational modeling of cognitive aspects. If we know something about systems engineering is that ad-hoc systems can always be built —or at least designed— to generate any kind of causal behavior; but the only thing they say about other systems with the same behavior is just this: they have the same behavior. Nothing more —but also, nothing less.

Behavior-based approaches to studying emotions are insufficient; but comparing the performance of the robotic model to animal behavior (decorticated animal performance on conditioning tasks) may be an *useful starting point*. Care is necessary however, because it is too easy to read meaning into particular behavior patterns. We would suggest, therefore, that there needs to also be some dynamics that involves agent internal states that can be related, in some way, to the real-world-relevant behavior and both of these aspects are indispensable for studying emotions [4].

The rigorous way of doing this discounting is the putting of all them — animals, animats, robots, simulations or embedded systems— inside a *single framework* for understanding that is both *common* and *rigorous*. Let's state some basement and brickery for this *Autonomous Systems Framework* (ASys).

4 Understanding Autonomy, Cognition and Emotion

A critical question in the research for bioinspired architectures for autonomy is the integration strategy for the multiple autonomic, emotional and cognitive systems that compose the control systems of a biological agent and possibly an artificial one. These three aspects —autonomic, cognitive, emotional— shall be clarified in oder to address the architecture from a holistic perspective.

4.1 Autonomy

Autonomy —in the bioinspired robotics context— can be understood in two major senses: 1) as the capability of a system —an animal, a robot— to manage its own resources in the pursue of its own goals; and 2) as the collection of minute regulatory capabilities that the autonomic systems offer in organisms.

This last collection of minimal capabilities —as those provided by the autonomic nervous system— is what we try to understand in their integration with cognitive and emotional generators of behavior in animals. We expect to understand them up to the point of been able to use this knowledge to build robots better fulfilling a set of requirements (be these mimicking a rat or provide better protection to vehicle passengers in a collision).

We will use the term *autonomic* to refer to these basic regulatory capabilities while reserving the term *autonomy* to refer to the highly desirable, system-level property emerging from the integration of the cognitive, emotional and autonomic systems.

4.2 Cognition

While there are plenty of suitable definitions of cognition for any imaginable purpose —vision, language, action, learning, *etc.* — there is also a somewhat post-modern relativism in assuming the fact that we can't make any progress towards a consolidated understanding of it. However, the very mechanics of our biological mind and the modes of operation of science —a form of socialized cognition— forces us to think that a convergence of the different understandings of cognition is not only possible but desirable, needed and, in fact, unavoidable.

The question is then what is the very mechanics of cognition that forces the convergence of its many aspects? From the general systems analysis that we are doing about the heterogeneous aspects of cognitive phenomena the point of convergence can be synthesized in a very compact sentence: *Cognition is model-based control*.

Let's expand a bit such a crude affirmation. Maturana and Varela, in his broadly referenced text on autopoiesis [5] end up equating life and cognition on the basis of a semiotical closure. This vision comes close to the understanding that both life and cognition are phenomena of profiting from information in the

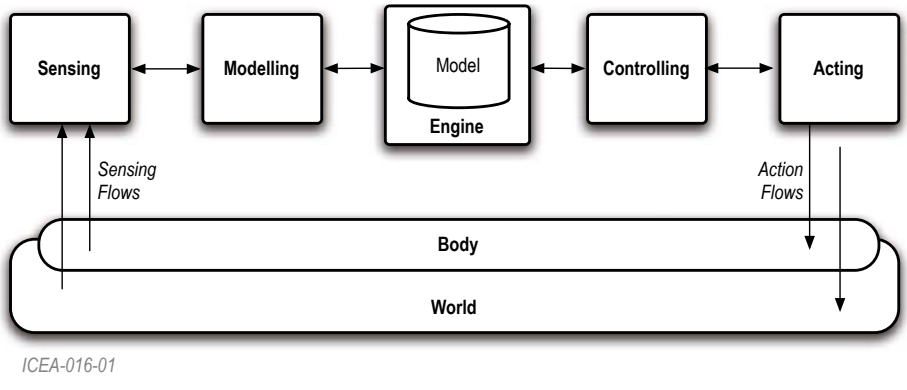


Fig. 2. A cognitive system realizes an epistemic loop —every good cognitive agent interacting with a system must contain a model of that system

environment —in the Batesonian sense of difference that makes a difference— by means of information-driven structures that get some profit from the information so input to the system.

In general, we can say that if a robot A is going to make some profit from its interaction with another agent B —whether natural or artificial— the way of maximizing the profit by A is the internalization of a model of B that can drive the interaction. This is well known in control system theory: the best controller of a plant is an inverse model of it. Paraphrasing the words of Conant and Ashby, we can say that every good cognitive agent in social interaction with another agent must contain a model of that other agent to drive their interaction [6].

This agent can be a cognitive agent of similar nature but can also be any entity in the universe of the agent that interacts with it. This absolute generality is exemplified in Figure 2 where the core cognitive pattern is constituted by what can be called an *epistemic loop*, where the knowledge the agent has is equated with the models it uses. This knowledge used to drive the agent's behavior must obviously include the world but also the self and any other agents for social performance.

4.3 Emotion

A general theory on the nature of emotion —esp. social ones— shall be a major result of the analysis of biological architectures in social settings. The question here is not to have a catalogue of specific emotions realizations to make our robot emote *à la* human (sad, happy, distressed, upset, sorry, grief, despair, depressed, joy, contended, pleased, delighted, excited, anger, indignation, shocked, appalled, hurt, remorse, guilt, ashamed, humiliated, embarrassed, prideful, triumphant, *etc.*). Instead, emotions shall be defined in terms of universal architectural primitives related to autonomous systems organization and autonomous generation of behaviors [7]. We must stress the fact that emotions are not not just for social display but that have an intrinsic, necessary, social function.

A core source for inspiration in the construction of architectures for autonomy and socialization are the architectures for emotion found in mammal brains and hormonal systems (see Figure 3). The systemic analysis approach leading to bioinspired robots is centered around the inputs that neuroscience and neuropsychology can provide for this theoretical endeavor and the degree to which models of neural substrates of integrated behavioral systems can contribute to the abstract picture of the nature of emotions in a larger architecture.

As an example, section 4.5 contains a brief depiction of a model of fear as of [9]. This is the class of information we must start from to derive general theories on integration of cognition, emotion and autonomic aspects.

4.4 Models of Cognitive Emotion: Shame as Example

In social settings, the so-called social emotions play a distinctive role that shall be leveraged in the construction of robots. Figure 4 shows a class diagram —with sub-classing relations— of some of the emotions analyzed in [10] and [7].

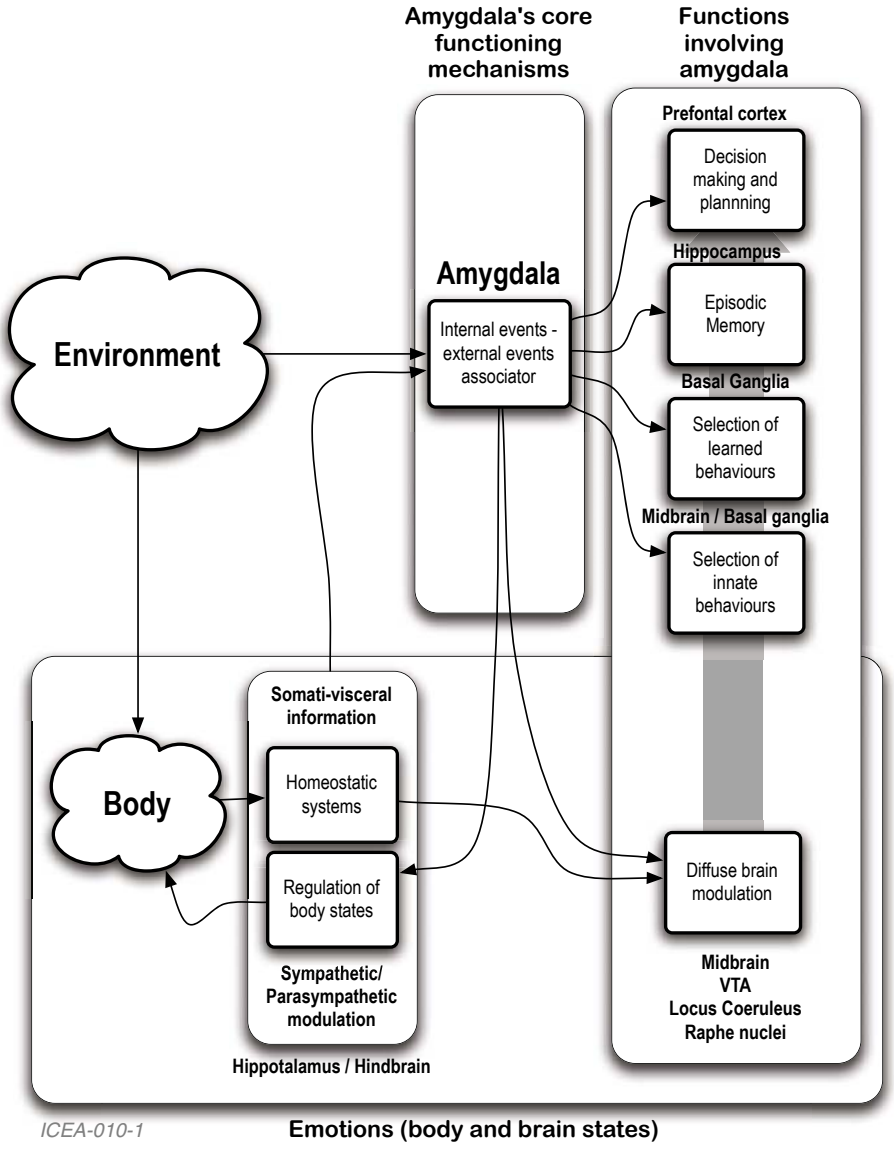


Fig. 3. Summary schematics showing the signal flow and functional/topographical allocation of main roles played by the amygdala. Notice the modeling —core associative— mechanisms implemented by the amygdala, that constitute a core assets of many other bodily functions also by means of the role that the amygdala plays in the modulation of emotions in terms of the control of broadcasted —diffused— brain states and bodily states [8].

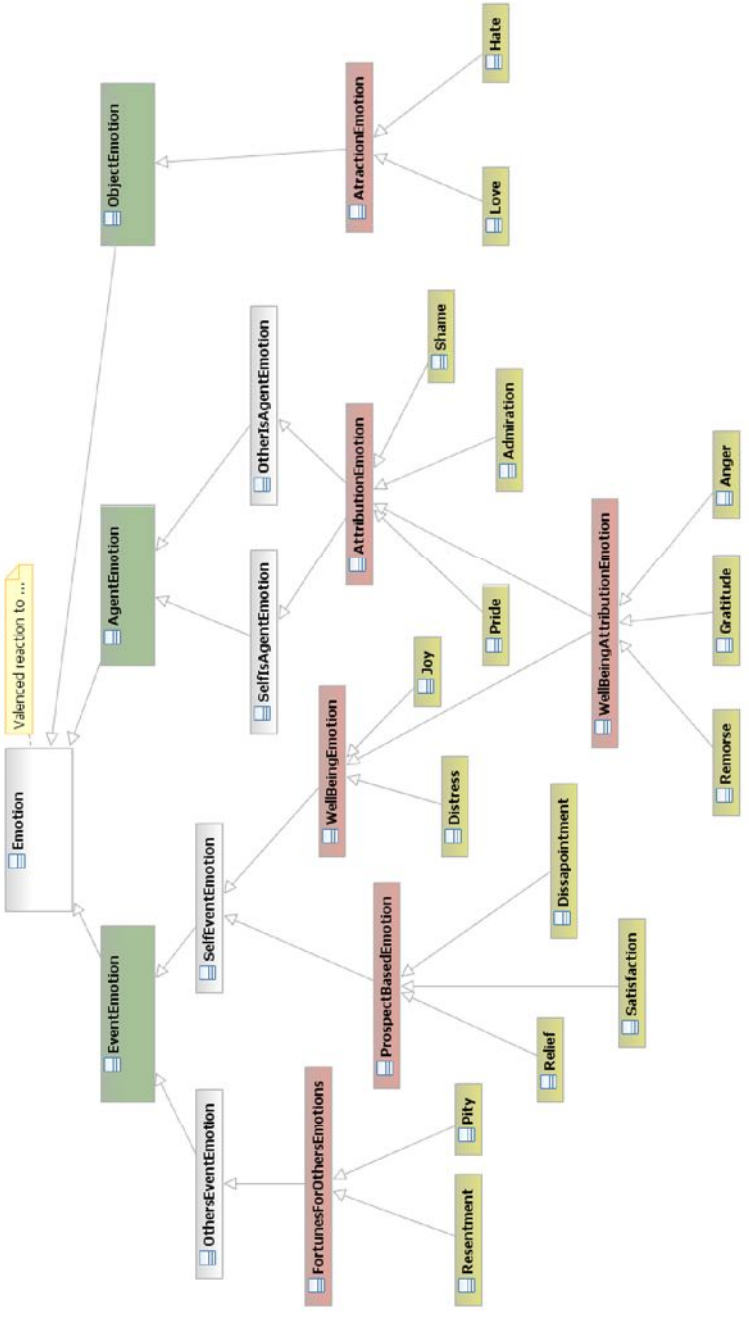


Fig. 4. A summary classification of cognitive emotion types following [10]. Green classes are the three core emotion source aspects —event, agent, object— and the red boxes indicate emotion types.

These emotions —for example *shame*— are part of the collection of higher level emotions triggered by cognitive —conscious— processing.

The characterization of the emotion types is done in terms of the operational aspects of agent functioning in relation with three classes of things: objects around the agent, events affecting the agent and agents causing the events.

When an agent is focused on *events* this is so because events can have a certain value for the agent in terms of the agent goal structure. An agent may be interested in a certain *object* because some of its properties are of relevance to the agent. And it may be interested in other *agents* because they may be the cause of events and object properties change.

This trichotomy —agent, object, event— fits the general vision of cognition as model based behavior and the analysis of the emotion types done by Ortony fits the different situation that can happen in terms of valenced outcome coming from actions derived from these models.

Of special importance, these models of agency over objects and events may render insights on higher cognitive-emotional, social behavior. For example, [11] is quite thought-provoking, highlighting the importance of representing the actions and intuiting the future acts of other agents, perhaps employing representations of the other agents' motivations. This could be very useful in creating levels of meta-consciousness wherein the agent has a framework to objectify itself. This will clarify issues regarding social action expectancies and fulfillments that trigger high-level cognitive emotions like shame.

4.5 Models of Bodily Emotion: Fear as Example

Sub-cognitive emotions —fear, hunger, thirst, *etc.* — are triggered by sub-cognitive states of the agent (even when they reach the conscious level by means of their effects and experience). The sub-cognitive mechanisms are studied at the level of the role that brain structures play in the onset and progression of emotive behavior in organisms. Of maximal importance in these models is the connectivity between structures due to the relative opacity to analysis to the structures themselves.

For example, in LeDoux's theoretical model of auditory fear conditioning [12], the amygdala neurons receive fast and broadly tuned inputs from the thalamus, and slower and more refined inputs from the cortex. The study of the conditioned auditory fear requires the hypothesizing of models of connections structures and the evaluation of the models in terms of predictability of operational effects.

Figure 5 shows the overall model used by Armony and co-workers to build a computational realization of the mechanisms for (conditioned) auditory fear processing. The computational model was adjusted —using synaptic connections parameters— so that each amygdala neuron learned to represent a restricted range of tone frequencies that together cover the audible spectrum.

This model realization had very interesting characteristics:

- Amygdala neurons did autonomous retuning to frequencies that were paired with fear elicitation. —later confirmed experimentally by recording from the amygdala of an animal that was fear-conditioned to auditory tones.

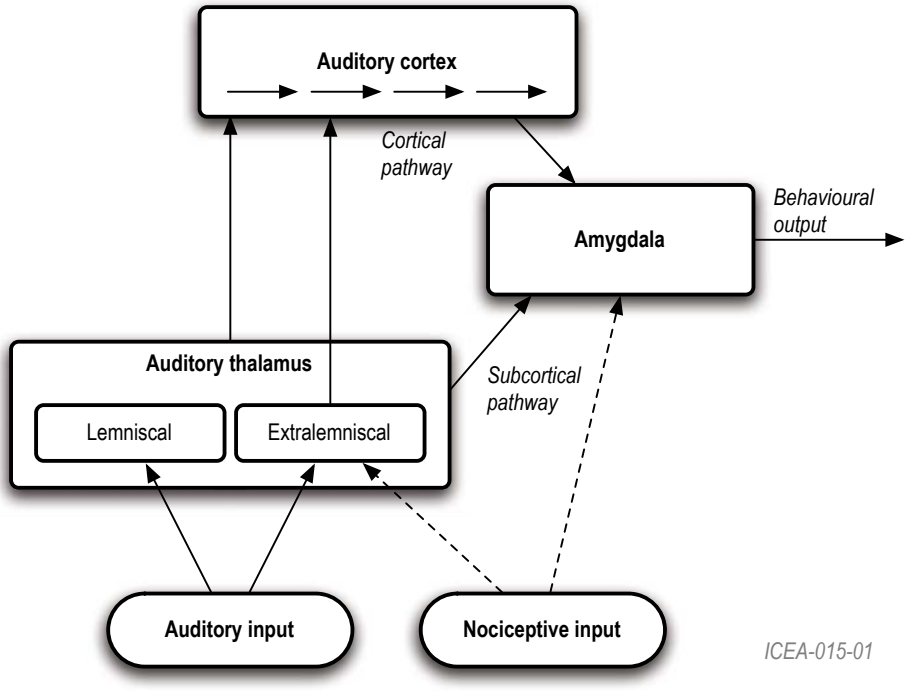


Fig. 5. Armony's model of auditory fear processing inspired in LeDoux's conceptual model of auditory fear conditioning [9]

– The model also showed that the reduction of cortical inputs to these cells did not totally hamper their capability for stimulus preference — again later confirmed experimentally by means of lesion studies.

In a sense, this model was a good model of fear processing in the rat at the neuronal level because it was able to make predictions that were experimentally testable (with success in this case).

The work to do upon this type of models is to determine what kind of general function the architecture provides or demonstrates. In this concrete case, the model —and the derived experimentation— provided support to the idea of dual routes of fearful stimulus processing. The type of questions we do must formulate are for example: Is this just a realization of a redundancy pattern? Is this a necessary pattern in structures supporting conditioned behavior? Work that addresses these questions with respect to the model of Armony and coworkers has recently been submitted for publication by [13].

To be able to answer these questions and reach some general conclusions we need to set a proper analysis framework for them. The following two sections address some of these general results —still pending for a more rigorous statement.

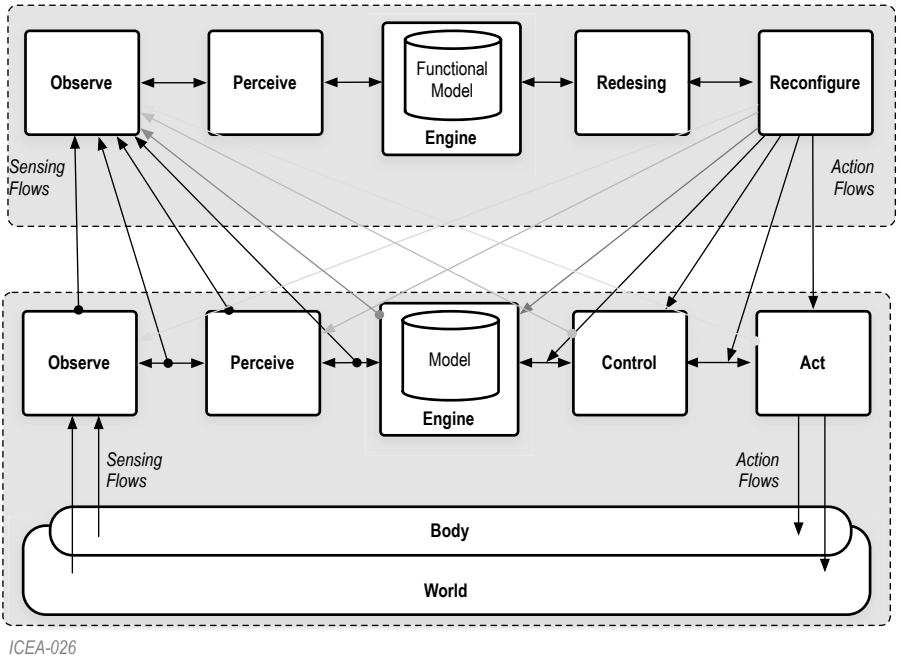


Fig. 6. The emotional meta-controller reshapes the functional organization of the systems. The externalization of the meta-controller state constitutes the grounding of the emotions to be displayed in social contexts to make other agents activate specific agent models concerning the displayer.

4.6 What Emotions Are Not

From the analysis of the several points of view considered in different works on emotion we have reached some conclusions regarding what emotions are not:

- They are not just sophisticated input handling — Not just reacting to bears.
- They are not just sophisticated action generation for social affective behavior — Not just showing embarrassment or shame.
- They are not just regoaling — Not just deciding to go to eat when doing sex.

4.7 What Emotions Do

An we have also reached some conclusions about how they work:

- Emotions do generate synthetic compact states (performing state space reduction) for the effective tuning and use of evolutionary meta-controllers.
- Emotions do change the control structures / component functions (patterns and roles) of subsystems.
- All this in a global controller configuration approach: transversal structural feedback.

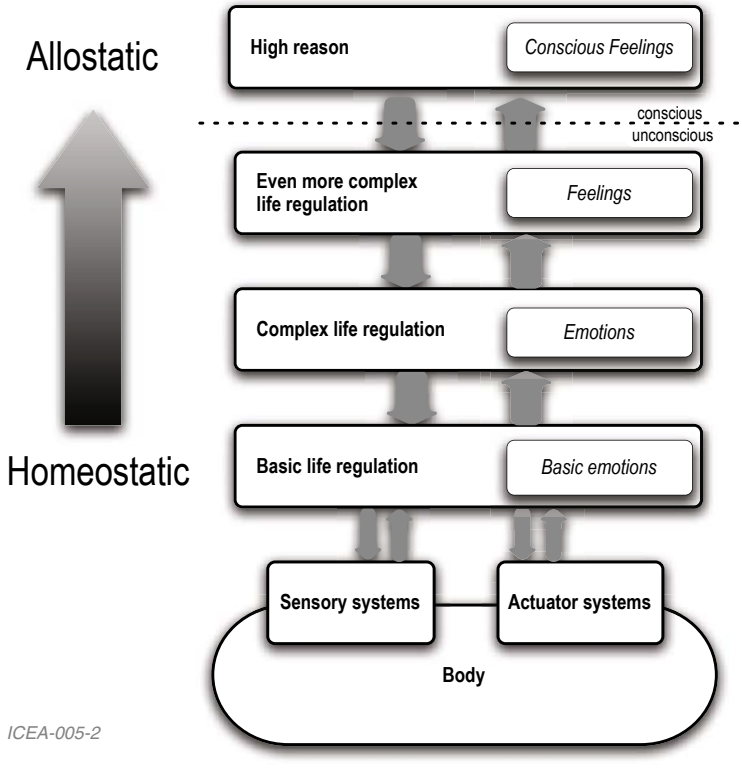


Fig. 7. A re-interpretation of Damasio's view [14]. Biological control systems — feedback or feedforward— happen in many layers and all of them incorporate emotional systems as aspects.

This perspective goes beyond Damasio's approach [14] of putting emotions/feelings as additional layers in hierarchical controllers. Figure 7 shows a re-interpretation of Damasio's view, where the multiply layered control systems incorporate emotional systems as aspects —in the computer science sense. This is also consistent with the allostatic/homeostatic regulation mechanisms to be described in [15].

Biological control systems of all kinds happen in many layers and all of them may incorporate emotional systems as transversal aspects.

5 Emotion as Structural Feedback

The global controller configuration mechanism provided by emotional systems operates transversally to configure control paths (sensing-processing-acting), changing operational modes of the whole system (goals-functions-patterns).

The performance of a multi-component control system, as is the case of the multifarious brain mechanisms, is based of the signal flow across components composing a *pattern* [16] (see Figure 8).

It is the pattern —structure + components playing the roles— what determines the function beyond the collection of individual atomic functions of the components. This is indeed the target of research focus of systems neuroscience (with systems biology at the backstage).

The three core aspects of emotion mechanisms can be analyzed in terms of stateful functions:

- Input: perception of relations agent-world (consider for example the role of amygdala; see Figure 3).
- Output: modification of organization (consider for example the role of the basal ganglia [17]).
- State: dispositions of the emotional system (in relation with somatic and mental states).

What is special about emotion as a control function is not just that it is meta-control (most cognitive systems are also meta-controllers); what is special is the fact that it is meta-control in a different range: emotional controllers do not target system state but system function. They are not just bodily regulation, nor bodily control but functional control.

Emotion actions *do change how the machine works* (they are causal entailments in Rosen's terms [18, 19]) implementing mechanisms for structural feedback [20].

This is much in line with the concept of emotion used in the component process model emotion appraisal analysis framework [21], where:

> *"... emotion is defined as an episode of interrelated, synchronized changes in the states of all or most of the five organismic subsystems in response to the evaluation of an external or internal stimulus event as relevant to major concerns of the organism."*

Figure 8 shows an abstract example of a structural functional change at the distributed pattern level. The initial pattern of components providing a concrete function (curvy blue arrowed components on the left) disappears and a new pattern appears (curvy blue arrowed components on the right). This new pattern of component interaction will possibly provide a somewhat different function that is better adapted to the present state of affairs.

6 Emotion as Transversal Integration Mechanics

Following basic principles of systems analysis we reach the conclusion that non-cognitive systems —*e.g.* homeostatic systems in animals— are degenerate cases of the broader class of cognitive systems understood as model-based behavioral systems [6, 22].

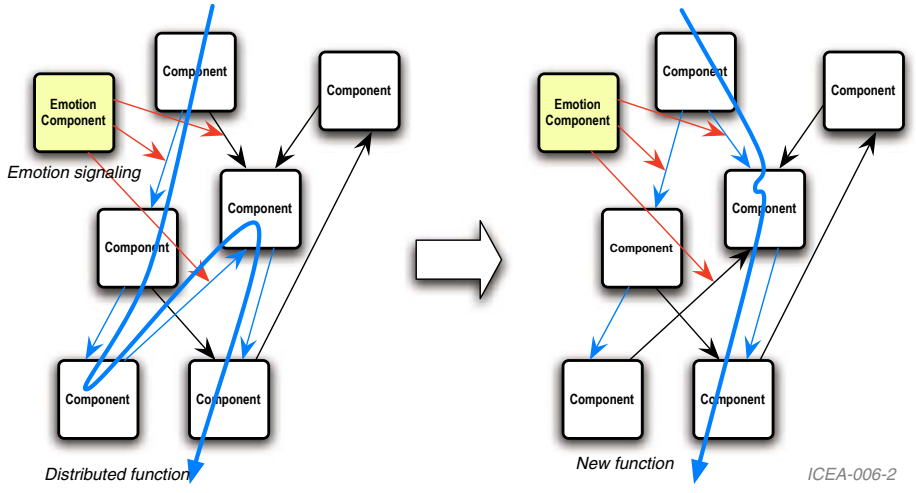


Fig. 8. An example of a structural functional change at the distributed pattern level. The pattern of components providing a function disappears and a new pattern appears (possibly providing a somewhat different function).

The global structure of a complex cognitive agent will be a collection of information processing elements, linked by information forwarding elements layered atop physical/information interfaces.

A global picture of this model is shown in Figure 9. Wrapping up this perspective on emotion we can summarize in the following conclusions:

- Emotions are transversal meta-control structures (up to the whole system level).
- Emotion range is system organization.
- Emotional states are synthetic, reduced dimensionality states receiving different names at different levels.
- The components of emotional systems are observers + sub-cognitive controllers + actuators.
- Displayed emotions are functional state externalizations that help model-based behavior coordination in social groups.

In the case of natural organisms, a core mechanism for the implementation of this transversal self-organization mechanics is neuromodulation: It permits the agent to shift between several different functional states depending upon state of danger alert, current assessment of state, needs and motivations, environmental factors (sensory such as night/day, complexity, reliability of current orienting estimates).

As of technical implementations of these mechanisms, Avila-Garcia and Cañamero [23] are simulating neuromodulation "by large-scale broadcast signals

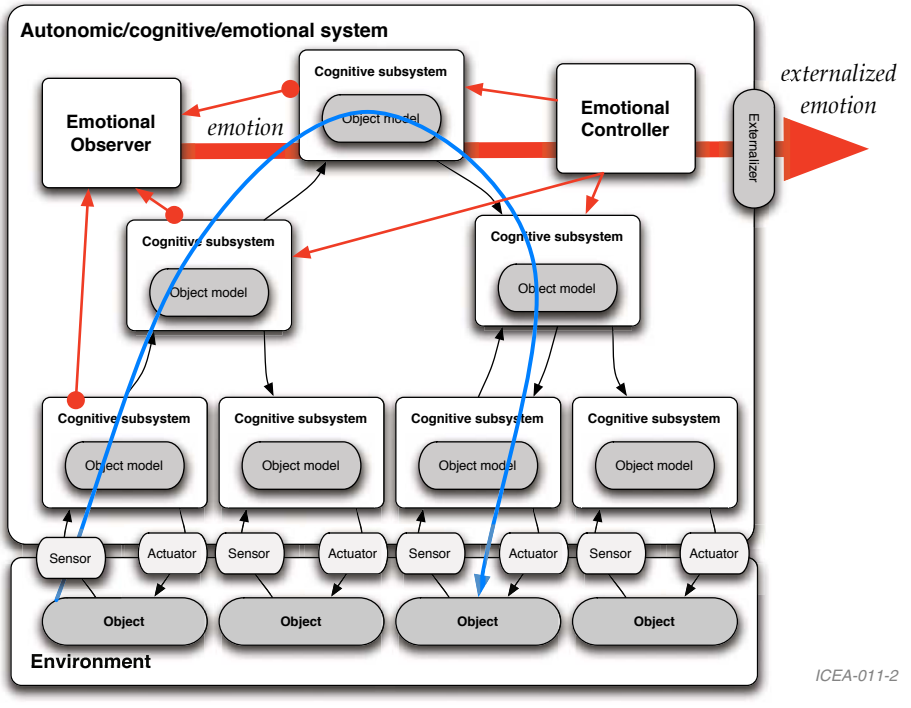


Fig. 9. Summary of emotion mechanics as distributed, layered, transversal meta-control

that have specific and local effects on their computational targets, and some of these effects could yield behavioral states analogous to emotional states."

What is even more interesting is that this class of transversal integration mechanism provides a cohesive structure that renders a multi-scale self-awareness mechanics that follow the design principles for machine consciousness described in [24].

7 The Importance of Function and Structure

The application of the model of emotion described in the former sections requires the implementation of a system including the necessary metacognitive competences to reflect and self-organize in terms of its very own functions. To do this, it is necessary to formulate a clear vision of the ontology underlying the engineering processes that generate the system and how these —ontology and processes— can be embedded in the inner mechanics of the cognitive agent.

7.1 Engineering Strategies

It is important to remember that in order to realize a system in an engineering or construction process there are three basic strategies —rigorous, artisan, evolution— (see section) that coalesce into two big classes:

- Systematic design — that may be supported by a systematic process or by automated tools realizing the systematic process.
- Evolutionary design —maybe evolutionary artisanal— that is not driven by inverse problem solving from requirements.

Intentional systematic design follows a rigorous approach from requirements to design using a systems engineering process.

Evolutionary —intentional or non non-intentional— are exploratory approaches based on partial knowledge and testing.

These two alternatives do have two major analytical strategies:

- Use a functional decomposition: rigorous in the case of systematic design and non-rigorous in the case of artisanal design.
- Not use a functional decomposition: while it is clear that evolution favors modularity, evolutionary advantage can give rise to cross-functional aspects across modules.

7.2 Functional Decomposition

Functional decompositions have the property of interface minimization and that is what enables both artisanal approaches and formal analyses. That implies that the dimension of the design space in a functional approach is at the minimum of all the alternatives. This is a major reason for humans following the functional strategy when doing intentional design. It simplifies design space exploration and system construction by the interface minimization property.

The other reason is a question of executability of models —mental or computational. Functional models are maximally cohesive —that is the reason for the minimization of the interfaces— and hence it is easier to master the emergent behavior from a system of functions because the interactions are minimal [25]. The proof of this comes from several sources but just considering system-level predictability when interfaces are being minimized we discover that at the end, when interfaces reach null, there's no interaction and the prediction of the system is just the union of the atomic predictions of each component. No emergent behavior increases the difficulty of the prediction task.

Functional separability, in general, does not happen in natural neural systems where otherwise apparently necessarily separated systems happen to be strongly connected. This may be conjunctural or may have a deeper implications concerning the core ontology and functioning of the brain. See for example the analysis of the separability and order of perception and action mechanisms in the visual system done by Rizzolati and Gallese [26, p.383] that concludes saying that action must precede —and is necessary for— perception.

Some researchers —inspired by ontogenic and filogenic emergence— expect that embedding the system with capabilities of self-organization will render a system that will be maximally operative without the burden and potential or apparent limitations of functionalisation. This has been proved as viable in engineering temporal scales in problems of reduced dimension, but as the general problem of design space exploration is an NP-complete problem, this strategy does not scale.

Hence, we conclude that the only viable alternative is the reduction of the dimensionality of the search space and the maximal reduction is achieved by functionalisation —or, to be more precise, by interface minimization.

7.3 Function and Autonomy

This last analysis let us state the central problem of how autonomy emerges from the integration of the cognitive, emotional and autonomic systems in strict functional terms: *autonomy is the closure of functional dependency.*

Cognitive, emotional and autonomic systems provide the required functions for a certain behavior (adapted, functional, *etc.*) by means of certain architectural patterns. To make a system autonomous we must functionally close the set of functions necessary for the fulfillment of the mission.

The difficulties of this work stem from the fact that our focus must go from the analysis of closeness in relatively simple and well-understood cases of small, relatively simple natural systems —protozoa, neurons— to those which really interest us —complex machines and higher organisms— going up the ladder of *specification hierarchies* proposed by Salthe [27]. The issues concerning function and, in particular, modularized function are at the forefront of semiosis and the grounding of meaning in the physical embodiment of agents.

Let's quote Josslyn [28] at this point:

> *"What characterizes these systems is that they involve processes of perception, interpretation, decision, and action with their environments. These semiotic processes involve the reference and interpretation of sign tokens maintained in coding relations with their interpretants. Thus issues arise here concerning the use and interpretation of symbols, representations, and/or internal models (whether explicit or implicit) by the system; and the syntactic, semantic, and pragmatic relations among the sign tokens, their interpretations, and their use or function for the systems in question."*

However, we must always bear in mind that semiosis is, in a sense, a maximally contingent phenomenon; so we must hold ourselves while doing the theorization in the position os escaping the *trap of necessity* so nicely set by biologism.

Let's continue quoting Josslyn:

> *"Semiotic relations are characterized by being contingent functional entailments. In particular, they are entailments, meaning regularities of*

> *constraints in system relations; which are functional, meaning deterministic (equivalent to a mathematical function); and which are contingent, namely that other such functional entailments (coding relations) could have been possible. This concept captures the arbitrary coding nature of symbol systems: the symbol and its referent share no properties in common except that the symbol refers to its referent when interpreted by an agent acting within the constraints of the symbol system. These are contrasted with purely physical systems, which are characterized by necessary functional entailments."*

We must analyze to what extent the natural world and the world of artificiality [29] can indeed share architectures because, in a first analysis, it may seem that the organisms are fully functionally closed while technical systems are not for the simple reason they must provide some externalized function —otherwise we would not expend energy into building them. However, a multi-scale analysis will show that from the social perspective biological organisms are not functionally closed. Individuals provide functions at the society level that revert into societal adaptation and eventually, by down-casting, into individual adaptation.

These survivability implications of function are not just issues of biology or theoretical cognitive science, and are becoming issues of relevance in "normal" —in Kuhnian terms— engineering activities. Consider for example the design for adaptation strategy for achieving advanced sustainable designs [30]. This methodology is based on the hypothesis that product life ends because a product is unable to adapt to change. Adaptability is becoming a critical issue in product line engineering, the artificial counterpart of biological species.

8 Next Steps

The next —ongoing— step is the consolidation of this ASys Vision in two directions:

- The formalization of the vision in mathematical terms employing the general systems theory modeling concept [31, 32] with an emphasis on systems engineering aspects [33] and exploring the possibilities of the more abstract category theory [34]. This indeed will render not only a formal theory of the integration of autonomic, emotional and cognitive aspects but a necessary formalization of each one of these aspects.
- The second way of consolidation is related to the use of formal and semiformal models in the implementation of technical systems —mobile robots and a process control testbed— that the ASLab members are implementing to test the theories described.

The case of formalization of autonomic aspects has already been done in the context of biological cybernetics [35] —plain systems theory applied to biological processes. Some attempts into formalization of theories of Damasio are available [36]. But while these may be interesting for the formalization of all aspects of

Damasio's theory —exposing and rendering more concrete some ambiguities— the concrete formalizations are always done using modeling simplifications that may render the formalization useless for some purposes. It is the case of the mentioned [36] that uses a discrete time logic that may not suit the dynamical analysis needed for the implementation of real-world social robots. Nonetheless this should be useful as a prototype and model.

The approach we are using for the implementation of social robot systems is the use of a model-driven engineering process based on semi-rigorous SysML models [37]. SysML is a systems modeling language with four pillars: *structure* and *behavior* ideally suited to capture system-level brain models; and *requirements* and *parametrics* for its transference into the technological implementations that are guided by performance measures. SysML-based tools support model transformation into realized models to execute over concrete application environments. The target platform being used by our team is based on the OMG CORBA Component specification [38] much in the line of other robot control software frameworks like ORCA and OROCOS.

Acknowledgements

We acknowledge the support of the Spanish Ministry of Education and Science through grant *C3: Control Consciente Cognitivo* and the European Commission thorough Grant *ICEA: Integrating Cognition, Emotion and Autonomy.*

References

1. Sloman, A.: What are emotion theories about? In: AAAI Spring Symposium (March 2004)
2. Cassell, J., Bickmore, T., Campbell, L., Vilhjálmsson, H., Yan, H.: Human conversation as a system framework: designing embodied conversational agents. In: Embodied conversational agents, pp. 29–63. MIT Press, Cambridge (2000)
3. Breazeal, C., Takanishi, A., Kobayashi, T.: Social robots that interact with people. In: Siciliano, B., Khatib, O. (eds.) Springer Handbook or Robotics, pp. 1349–1369. Springer, Heidelberg (2008)
4. Morse, A., Lowe, R., Ziemke, T.: Towards an enactive cognitive architecture. In: International Conference on Cognitive Systems, CogSys 2008 (Accepted) (2008)
5. Maturana, H.R., Varela, F.G.: De Máquinas y Seres Vivos. Editorial Universitaria, Chile (1972)
6. Conant, R.C., Ashby, W.R.: Every good regulator of a system must be a model of that system. International Journal of Systems Science 1(2), 89–97 (1970)
7. Wierzbicka, A.: Defining emotion concepts. Cognitive Science 16, 539–581 (1992)
8. Mannella, F., Zappacosta, S., Mirolli, M., Baldassarre, G.: D22 the role of amygdala in the regulation of motivation, emotion and behaviour. Technical Report 22, The ICEA Consortium (2006)
9. Armony, J., Servan-Schreiber, D., Romanski, L., Cohen, J., LeDoux, J.: Stimulus generalization of fear responses: effects of auditory cortex lesions in a computational model and in rats. Cerebral Cortex 7, 157–165 (1997)

10. Ortony, A., Clore, G.L., Collins, A.: The Cognitive Structure of Emotions. Cambridge University Press, Cambridge (1988)
11. Jeannerod, M.: How do we decipher other's emotions? In: Fellous, J.M., Arbib, M. (eds.) Who needs emotions? Oxford University Press, Oxford (2005)
12. LeDoux, J.: The Emotional Brain. Simon & Schuster, New York (1996)
13. Lowe, R., Humphries, M., Ziemke, T.: Evaluating the structural composition of a neurocomputational model of fear conditioning. Connection Science (Submitted) (2008)
14. Damasio, A.: The Feeling of What Happens: Body, Emotion and the Making of Consciousness. Vintage (2000)
15. Lowe, R., Morse, A., Ziemke, T.: An enactive approach for modelling cognition, emotions and autonomy: Predictive regulation at different levels of organizational complexity. Adaptive Behavior (to be Submitted) (2008)
16. Sanz, R., Segarra, M.J., de Antonio, A., Matía, F., Jiménez, A., Galán, R.: Design patterns in intelligent control systems. In: Proceedings of IFAC 14th World Congress, Beijing, China (1999)
17. Humphries, M.D., Stewart, R.D., Gurney, K.N.: A physiologically plausible model of action selection and oscillatory activity in the basal ganglia. The Journal of Neuroscience 26(50), 12921–12942 (2006)
18. Rosen, R.: Anticipatory Systems. Pergamon Press, Oxford (1985)
19. Rosen, R.: Life Itself: A Comprehensive Inquiry into the Nature, Origin, and Fabrication of Life. Columbia University Press (1991)
20. Sanz, R., López, I.: Minds, MIPS and structural feedback. In: Performance Metrics for Intelligent Systems, PerMIS 2000, Gaithersburg, USA (2000)
21. Sander, D., Grandjean, D., Scherer, K.R.: A systems approach to appraisal mechanisms in emotion. Neural Networks 18, 317–352 (2005)
22. Sanz, R., Gómez, J., Hernández, C., Alarcón, I.: Thinking with the body: Towards hierarchical scalable cognition. In: Handbook of Cognitive Science: An Embodied Approach. Elsevier, Amsterdam (2008)
23. Avila-Garcia, O., Cañamero, L.: Using hormonal feedback to modulate action selection in a competitive scenario. In: Schaal, S., Ijspeert, A., Billard, A., Vijayakumar, S., Hallam, J., Meyer, J.A. (eds.) From Animals to Animats 8: Proc. 8th Intl. Conf. on Simulation of Adaptive Behavior (SAB 2004), pp. 243–252. MIT Press, Cambridge (2004)
24. Sanz, R., López, I., Rodríguez, M., Hernández, C.: Principles for consciousness in integrated cognitive control. Neural Networks 20(9), 938–946 (2007)
25. Baylin, E.: Functional Modeling of Systems. Gordon and Breach, New York (1999)
26. Rizzolatti, G., Gallese, V.: Do perception and action result from different brain circuits? the three visual systems hypothesis. In: van Hemmen, J.L., Sejnovski, T.J. (eds.) 23 Problems in Systems Neuroscience. Oxford University Press, Oxford (2006)
27. Salthe, S.: Development and Evolution. MIT Press, Cambridge (1993)
28. Joslyn, C.: Levels of control and closure in complex semiotic systems. In: Workshop on Closure: Emergent Organizations and their Dynamics (1999)
29. Simon, H.A.: The Sciences of the Artificial, 2nd edn. MIT Press, Cambridge (1981)
30. Kasarda, M.E., Terpenny, J.P., Inman, D., Precoda, K.R., Jelesko, J., Sahin, A., Park, J.: Design for adaptability (dfad)-a new concept for achieving sustainable design. Robotics and Computer-Integrated Manufacturing 23(6), 727–734 (2007)

31. Klir, G.C.: An approach to general systems Theory. Van Nostrand Reinhold (1969)
32. Mesarovic, M., Takahara, Y.: Abstract Systems Theory. Springer, Berlin (1989)
33. Wymore, A.W.: Model-Based Systems Engineering. CRC Press, Boca Raton (1993)
34. Lawvere, F.W., Schanuel, S.H.: Conceptual Mathematics: A first introduction to categories. Cambridge University Press, Cambridge (1997)
35. Bayliss, L.: Living Control Systems. The English University Press, London (1996)
36. Bosse, T., Jonker, C.M., Treur, J.: Formalisation of damasio's theory of emotion, feeling and core consciousness. Consciousness and Cognitio (In Press) (2008)
37. OMG: Omg SysML specification. OMG Adopted Specification ptc/06-05-04, Object Managemnt Group (2006)
38. OMG: Corba component model. release 4.0. Technical Report Document - formal/06-04-01, Object Management Group, Falls Church, USA (2006)

A New Dynamic Edge Detection toward Better Human-Robot Interaction

Abdul Rahman Hafiz, Fady Alnajjar, and Kazuyuki Murase

Graduate School of Engineering, University of Fukui, Fukui, Japan
{abdul,fady,murase}@synapse.his.fukui-u.ac.jp

Abstract. Robot's vision plays a significant role in human-robot interaction, e.g., face recognition, expression understanding, motion tracking, etc. Building a strong vision system for the robot, therefore, is one of the fundamental issues behind the success of such an interaction. Edge detection, which is known as the basic units for measuring the strength of any vision system, has recently been taken attention from many groups of robotic researchers. Most of the reported works surrounding this issue have been based on designing a static mask, which sequentially move through the pixels in the image to extract edges. Despite the success of these works, such statically could restrict the model's performance in some domains. Designing a dynamic mask by the inspiration from the basic principle of "retina", and which supported by a unique distribution of photoreceptor, therefore, could overcome this problem. A human-like robot (RobovieR-2) has been used to examine the validity of the proposed model. The experimental results show the validity of the model, and it is ability to offer a number of advantages to the robot, such as: accurate edge detection and better attention to the front user, which is a step towards human-robot interaction.

Keywords: Edge detection, human-robot interaction, biological inspired retina.

1 Introduction

Developing a human-like robot's controller, inspired from the principles of neuroscience, is one of the challenging tasks for many groups of robotics researchers [1]. The difficulty in such a system can be summed up into three main points as diagrammatically shown in Fig.1 : (A) A mechanism for human-robot interaction, which is, mainly relies on robot's vision, speech recognition, sensor-motor interaction, etc. (B) A mechanism for learning and memory, which gives the robot the feature of learn and/or teach. (C) A mechanism for homeostasis, which gives the robot a degree of an internal stability. In this study, we are highlighting the issue of enhancing the robot's vision toward better human-robot interaction. More precisely, we are introducing a new mechanism for dynamic edge detection that supported by variant photoreceptor distributions for better robot's vision.

Edge detection is classified as a fundamental step in many machine vision and image processing applications and systems [2],[3],[4]. The degree of its importance lies on the level of automation needed in the image processing system [5]. It is mainly

J.-H. Kim et al. (Eds.): FIRA 2009, LNCS 5744, pp. 44–52, 2009.

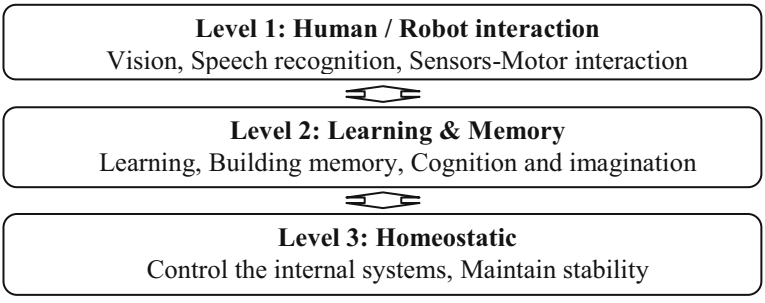


Fig. 1. Human-like robot's control system

responsible for extracting accurate edges from the image, which paves the image for any farther processes such as, object recognition, feature extraction, 3D environment constructing, etc. [6].

So far, many works have been done to develop a unique technique that can guarantees, to some degree, high quality edge detection with less noise and computational time [7]. Most of these works were basically relying on designing a static mask that sequentially moves through the pixels in the image toward extracting edges [8],[9]. Despite the success of these works, the idea of pre-designing the mask, however, could limit the performance of these models, especially when dealing with the real world applications.

In recent years, researchers have been investigating the biological concept of "retina" to try to overcome the above problem, since it is now widely accepted that the biological inspired technology is a powerful source to achieve better result, simple structure and less computational time [1].

Cognitive Vision Research Group in Hungarian Academy of Sciences [10], have design a model for edge detection based on the center surround structure receptive fields that present in retina. They have simulated the eye tremors and drifts to enhance the output image; however, their filter was static and cannot distinguish between the noise and the edge. In [11], authors have tried to solve the same problem by building a neural network and trained it by Back Propagation to ignore the noise. However, the work does not guarantee the ability to detect different edge formations.

Giorgio in [12] worked in an attention system for humanoid robot based on space variant vision for motor control. However, the work has missed the nature of the variety of photoreceptor in the human retina.

Along this line of research, we are here aiming to develop a dynamic mask inspired by the neuron connection of "retina". The proposed mask is constructed by artificial neural network and applied in parallel to the robot's view. Experimental results show the validity of the proposed model in achieving edge resolution in efficient manner that could lead to better human-robot interaction.

The following section highlights the biological concept of retina. Section 3, describes in details the proposed method. Section 4, shows experimental setup and results. Finally, section 5, concludes the work and gives the direction for a future work.

2 The Biological Concept of Retina

Retina is considered as a part of the brain. It is responsible for performing the first stage of the image processing, e.g., edge detection, before passing its signal to the vision cortex in the brain, for the farther processing [13]. As it can be seen in Fig.2a, the horizontal cell is between both the photoreceptors (cones) and bipolar cells. When the light is absent, the horizontal cell releases the neurotransmitter gamma-aminobutyric acid, which known as (GABA) receptors [14]. This phenomenon has an inhibitory effect on the photoreceptors. When the light is shone onto a photoreceptor, the photoreceptor hyperpolarizes and reduces the release of glutamic acid (glutamate) which excites the horizontal cell to reduce the release of GABA. This reduction of inhibition leads to a depolarization of the photoreceptors. This complicated process, however, is still a subject of hot debate in the community of retina scientist [15].

The functionality of the horizontal cell can be summarized by two main points Fig.2b, i- A single bipolar cell carries a fairly blurry response to its ganglion cell, which thereafter, transfers the signal to the brain. ii- Horizontal cells add an opponent signal that is spatially constrictive, and gives the bipolar cell the center surrounds representation.

According to [14], the reduction of GABA varies according to the brightness of the light that shines onto the photoreceptor and the time that this light is present, and that will give the center surround representation accurate information about the edges [14]. From the above phenomenon, we have got the idea of designing a dynamic mask that adapts itself based on the given input pattern, and applied it directly into the robot's vision.

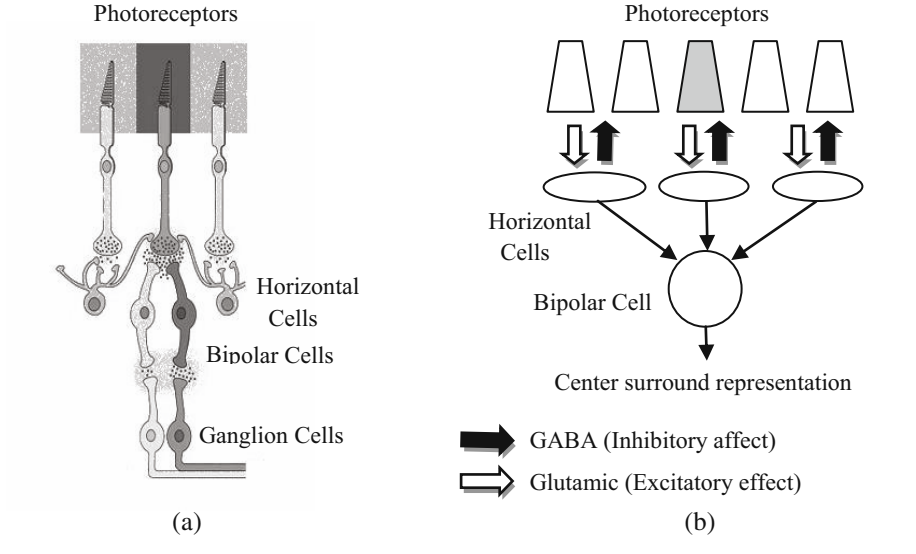


Fig. 2. (a) The structure of center surrounds neural connection in retina (adopted from [16]). (b) Edge Detection in retina.

3 The Proposed Model

3.1 Dynamic Edge Detection

To simulate the effect of horizontal cells in human retina, we designed a two-layer neural network mask Fig.3a. The input layer, which contains 9 neurons (3X3), used to represent the photoreceptor cells, while the output layer, which has one neuron, to represent the Horizontal cell. From the mathematical point of view, mapping between the input layer and the output layer can be expressed by Eq.1.

$$Output = (\sum_{i=1}^{3}\sum_{j=1}^{3} w_{i,j} * Input_{i,j}) / 9 \quad (1)$$

Where w represents the weight (GABA variation), and i ,j is the Location of the Input neuron.

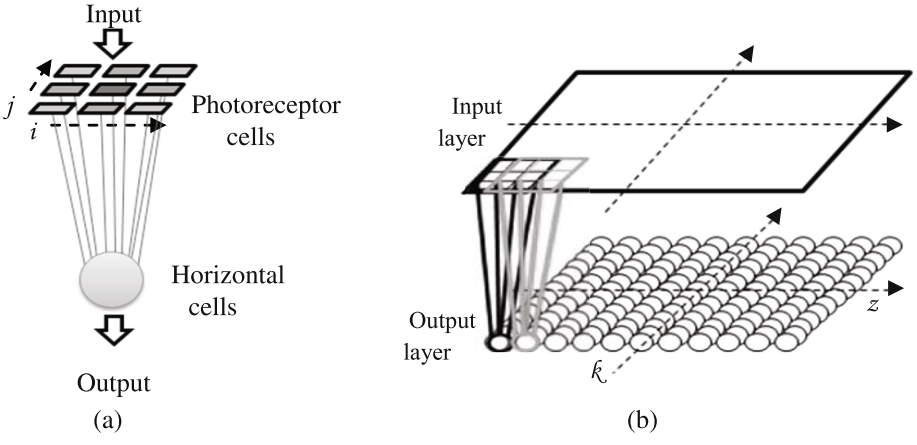


Fig. 3. (a) The proposed dynamic mask, (b) The whole network applied for edge detection

The above designed mask is part of a multi masks that represent in parallel the robot's vision Fig.3b. The address of any mask in the robot's vision is identified by the value of *k, z*. The weights in each mask are updated similar to that done by the *glutamates* and *GABA* effect [14]. The redistribution of the weights is done based on the contribution of each input pixel to the location *z,k,* Eq.2. Inputs with high/low contribution value gradually increased/decreased its related weights overtime Eq.3, Eq.4, Eq.5. Note that the overall summation of the weights in each mask should maintain its stability Eq.6.

We believe by the above proposed model, each mask can adapt itself during the time to reflect the input pattern of its related area.

$$C_{(z,k)i,j} = (w_{(z,k)i,j} * Input_{(z,k)i,j}) / output_{(z,k)} \quad (2)$$

$$W_{(z,k)} = \sum_{i=1}^{3} \sum_{j=1}^{3} (0.1 * w_{(z,k)i,j}) \tag{3}$$

$$\left(w_{(z,k)i,j}\right)_{(t)} = \left(w_{(z,k)i,j} - (0.1 * w_{(z,k)i,j})\right)_{(t-1)} \tag{4}$$

$$\left(w_{(z,k)i,j}\right)_{(t)} = \left(w_{(z,k)i,j} + (C_{(z,k)i,j} * W_{(z,k)})\right)_{(t-1)} \tag{5}$$

$$\left(\sum_{i=1}^{3} \sum_{j=1}^{3} w_{(z,k)i,j}\right)_{(t)} - \left(\sum_{i=1}^{3} \sum_{j=1}^{3} w_{(z,k)i,j}\right)_{(t-1)} = 0 \tag{6}$$

3.2 Variant Photoreceptor Distribution

Applying such a multi masks in parallel to the whole robot's view could in one side reduce the concentration of robots, and on the other side, cause high computational time and high noise. Therefore, variant distributions of the photoreceptors (cones and rods) are applied to the robot's view similar to that exist in retina [12] Fig.4. Where the cones are dense in the center of the retina fovea, and it is responsible for sharp and color vision, while rods are absent in fovea but dense elsewhere, and it is more sensitive to motion [11].

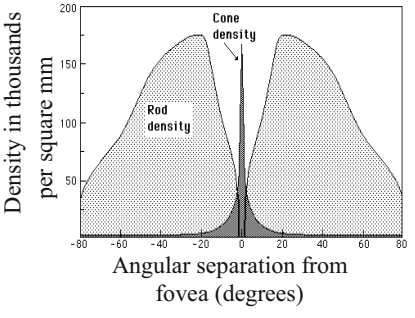


Fig. 4. The photoreceptor distribution in human retina

We, in here, duplicate similar arrangement into the robot's vision Fig.4. Photoreceptors in center of the image have more resolution and are responsible to edge detection and object recognition, while photoreceptors in the peripheral area have less resolution and are responsible for motion detection.

The movement of the robot's eyes (2 color cameras mounted in the RoboVie-R2's head with 2 degree of freedom each) has been applied to support the proposed model. The movement techniques can be compared by the one called (Saccade and Pursuit) [10]. This movement gives a robot wider view, and tries to maintain the main subject most of the time in the center of fovea.

The output of the variant photoreceptor distribution can be represented by (Eq.7).

$$Output = ED * e^{-(4*r)^2} + MD * (1 - e^{-(4*r)^2}) \quad (7)$$

Where the ED represents the Edge Detection, MD represents the Motion Detection, and r represents the distance from the center of the image Fig.5.

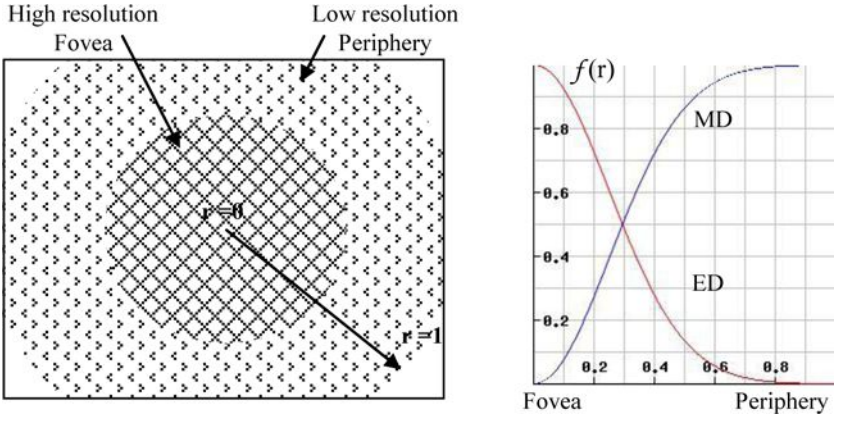


Fig. 5. Space variant in the robot's vision (300*200 pixels)

4 Experimental Results

4.1 Validity

In this stage, we examine the validity of the proposed dynamic model in extracting clear edges from real time robot's vision. The model was applied until all the weights in each mask were updated and reached to its highest value.

Fig.6 shows the process of edges' formation. It can be seen that within 2 seconds the edges in the image became clear.

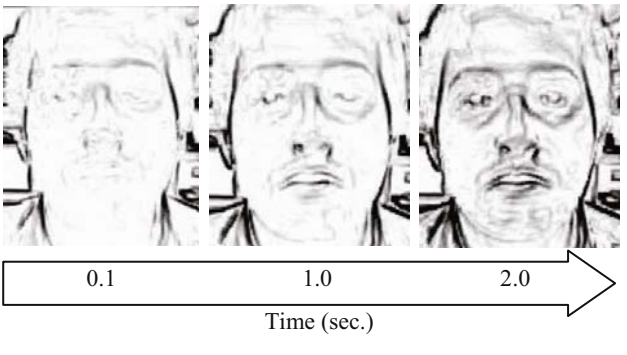


Fig. 6. The proposed network's output in real time (2 seconds)

We have compared our result with a model that used static mask edge detection, e.g. [8], see Fig.7. From the figure, we can observe that our proposed model can detect clearer edges in real time with less noise than that used static mask. It is interesting to say that the dynamic feature of our model gives the ability of each mask to adapt itself to suit within the input pattern.

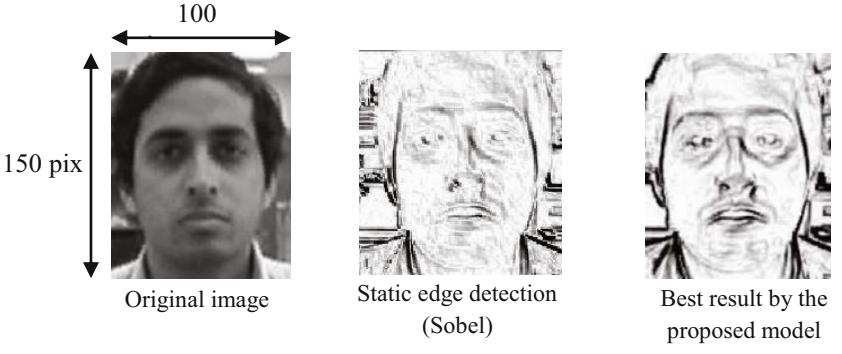


Fig. 7. Comparison between the proposed network and the static edge detection filters proposed by [8]

4.2 Better Human-Robot Interaction

As mentioned earlier, our main target is to build a human-like robot. In this experiment, in particular, we examine the ability of the proposed dynamic model in enhancing the robot's vision, which thereafter, can enhance human-robot interaction, the sub-target of our main target.

We have applied the above proposed model to the robot. The robot was presented into office-like environment with a number of students moving around. In addition to our model, the robot was run by simple face recognition program (developed in our lab), and speech recognition borrowed from the open source by Microsoft [17].

At the initial time, robot started to look around randomly. When one user gets close to the robot and started a conversation, the robot gave attention to the user by

Fig. 8. After Appling the proposed space variant in the robot vision

centering his face into Fovea and started to response to the user Fig.8. During the conversation the robot always tried to keep the user into Fovea, training its mask, as long as the user is giving it attention. The robot was also neglecting the other users who were moving around even that the robot was aware of them.

5 Conclusions

This paper addressed a problem of edge detection that can enhance the robot's vision in real world applications toward better human-robot interaction. The proposed model based on developing a dynamic mask inspired by the neuron connection of "retina". It is constructed by two-layer artificial neural network that is applied in parallel on the robot's vision. Synaptic weights, in each mask, were updated gradually during the time based on the founded edges in the image. To avoid the long computational time and to decrease the area of robot's main attention, two variant distributions of the photoreceptors were applied to the robot's vision. Edge detection processes lay more densely in the central region called fovea, while they are gradually sparser in the periphery, and vice versa for the motion detection.

The distributed nature of the network would also allow for a parallel implementation, making real-time frame-rate processing a definite possibility.

The two freedom movement of robot's eyes was also added to the model. The objective of such a movement lies in the attempt to keep the subject in a certain area in the fovea, so that, the weights in the mask maintain its strength, as well as, give wider area for robot's view.

Experimental results were focused to examine the validity of the proposed model in achieving edge detection in an efficient manner with less noise than those obtained by static model. We believe that our proposed method would be efficient for any dynamic application, where the moving edges require always detecting. The robot could give more attention to the main subject but at the same time keep aware of the potential targets that moving around. We believe that this could be a first step of human-robot interaction.

Besides that this study focus to build a model inspired from a biological concept toward new solutions to robotics but more importantly is the goal of gaining a better understanding of how the brain of living systems solves the same sort of problems.

Acknowledgments. This work was supported by grants to KM from Japanese Society for Promotion of Sciences and by the University of Fukui.

References

[1] Floreano, D., Mattiussi, C.: Bio-Inspired Artificial Intelligence: Theories, Methods, and Technologies. The MIT Press, Cambridge (2008)

[2] Manian, V., Vasquez, R.: Multiresolution edge detection algorithm applied to SAR images. In: IEEE Geoscience and Remote Sensing Symposium, vol. 2, pp. 1291–1293 (1999)

[3] Rydberg, A., Borgefors, G.: Extracting multispectral edges in satellite images over agricultural fields. In: Proceedings of International Conference on Image Analysis and Processing, pp. 786–791 (1999)
[4] Tewfik, A.H., Assaad, F.A., Deriche, M.: Edge detection using spectral estimation techniques. In: The 6th Multidimensional Signal Processing Workshop, pp. 34–35 (1989)
[5] Stewein, J.C., Ferris, T.L.J.: The asterisk operator. An edge detection operator addressing the problem of clean edges in bone X-ray images. In: Proceedings of the Second International Conference on Knowledge Based Intelligent Electronic Systems, vol. 3, pp. 28–31 (1998)
[6] Karantzalos, K., Argialas, D.: Improving Edge Detection And Watershed Segmentation With Anisotropic Diffusion and Morphological Levellings. International Journal of Remote Sensing 27, 5427–5434 (2006)
[7] Meylan, L., Alleysson, D., Susstrunk, S.: A Model of Retinal Local Adaptation for the Tone Mapping of Color Filter Array Images. Journal of the Optical Society of America A (JOSA A) 24, 2807–2816 (2007)
[8] Sobel, I., Feldman, G.: A 3x3 Isotropic Gradient Operator for Image Processing: presented at a talk at the Stanford Artificial Project (unpublished but often cited) (1968)
[9] Canny, J.: A computational approach to edge detection. IEEE Transactions on Pattern Analysis and Machine Intelligence 8, 679–698 (1986)
[10] Resko, B., Roka, A., Csapo, A., Baranyi, P.: Edge Detection Model Based on Involuntary Tremors and Drifts of the Eye. Journal of Advanced Computational Intelligence and Intelligent Informatics 11, 648–654 (2007)
[11] Becerikli, Y., Engin, H.: Alternative Neural Network Based Edge Detection. Neural Information Processing 10, 193–199 (2006)
[12] Metta, G.: An Attentional System for a Humanoid Robot Exploiting Space Variant Vision. In: IEEE-RAS International Conference on Humanoid Robots, Tokyo, Japan, pp. 359–366 (2001)
[13] Kolesnik, M., Barlit, A., Zubkov, E.: Iterative Tuning of Simple Cells for Contrast Invariant Edge Enhancement. In: Bülthoff, H.H., Lee, S.-W., Poggio, T.A., Wallraven, C. (eds.) BMCV 2002. LNCS, vol. 2525, pp. 27–37. Springer, Heidelberg (2002)
[14] Verweij, J., Hornstein, E.P., Schnapf, J.L.: Surround Antagonism in Macaque Cone Photoreceptors. The Journal of Neuroscience 268, 1053–1056 (2003)
[15] Dacey, D.: Primate Retina: Cell Types, Circuits and Colour Opponency. Progress in Retinal and Eye Research 19, 647–648 (2000)
[16] Institute of Neurosciences (INMHA), `http://thebrain.mcgill.ca`
[17] Microsoft, `http://www.microsoft.com/speech/default.mspx`

Face Recognition Using ALLE and SIFT for Human Robot Interaction

Yaozhang Pan, Shuzhi Sam Ge, and Hongsheng He

Social Robotics Lab, Interactive and Digital Medial Institute, Department of Electrical and Computer Engineering, National University of Singapore, 117576
Tel.: (65) 6516-6821; Fax. (65) 6779-1103
elegesz@nus.edu.sg
http://robotics.nus.edu.sg/sge/

Abstract. Face recognition is a very important aspect in developing human-robot interaction (HRI) for social robots. In this paper, an efficient face recognition algorithm is introduced for building intelligent robot vision system to recognize human faces. Dimension deduction algorithms locally linear embedding (LLE) and adaptive locally linear embedding (ALLE) and feature extraction algorithm scale-invariant feature transform (SIFT) are combined to form new methods called LLE-SIFT and ALLE-SIFT for finding compact and distinctive descriptors for face images. The new feature descriptors are demonstrated to have better performance in face recognition applications than standard SIFT descriptors, which shows that the proposed method is promising for developing robot vision system of face recognition.

1 Introduction

Face recognition is the one of the most popular research topics in pattern recognition during this decade [1,2,3]. It can be widely used in entertainment, information security, intelligent robotics and so on. Recently, great development has been done by researchers on both algorithm and system. A critical part in face recognition is the dimension reduction algorithm for feature extraction. In this area, global feature extraction algorithm such as principal component analysis (PCA) , linear discriminant analysis (LDA) and all the methods based on combination of this two gave many good results in applications on facial recognition. Later, as a nonlinear extension of PCA, Kernel PCA (KPCA) [4] has shown great advantages on data representation for clustering and classification of complicated facial data set. Based on the very observation that null subspace contains useful information for clustering, in [5], Lu et. al proposed kernel direct discriminant analysis (KDDA), which is combination of KPCA and direct linear discriminant analysis (DLDA). Another combination of LDA and KPCA, called Complete Kernel Fisher Discriminant (CKFD), has been proposed in [6]. All these kernel based methods have a major disadvantage in that the selection of kernel function and its parameters is usually made by trial and error or based on experience, which greatly weaken the practical value of these methods. Moreover, the final projections are related to all the training samples, so that the requirement for training samples are usually strict.

J.-H. Kim et al. (Eds.): FIRA 2009, LNCS 5744, pp. 53–62, 2009.

Compare to these kernel based methods, LLE [7] has its own advantages because of unsupervised property. On one hand, it do not need training samples, which is especially helpful for small sample size of the face pattern's distribution. On the other hand, it only has one simple parameter, K number of neighbors selected, to be chosen, which make it easy to be applied. However, the performance of original LLE will decline when the data distribution is not well or uniformed distributed. Another problem is that the algorithm is not robust to parameter changes. To combat these problems, WLLE is proposed to obtain better performance for complicated data set such as face images [8].

In this paper, we combine some feature extraction algorithms SIFT, LLE and its new extension, ALLE, to obtain reliable and robust face recognition algorithm.The new feature descriptors are demonstrated to have better performance in face recognition applications than standard SIFT descriptors, which shows the proposed method is a promising way for developing robot vision system of human recognition.

The main contributions of this paper are:

(i) Combination of adaptive neighborhood selection and LLE algorithm to form a new dimension reduction algorithm ALLE which are more robust and accurate;
(ii) Utilization of SIFT to obtain feature descriptors to do face recognition for human robot interaction; and
(iii) Combination between LLE/ALLE, SIFT and database manager is integrated into the face recognition module to achieve a precise recognition accuracy.

In the following sections, we will describe how each of these modules works in detail.

2 Face Recognition by LLE/ALLE-SIFT

In this part, face recognition module is to combine locally linear embedding (LLE) or adaptive locally linear embedding (ALLE) and scale-invariant feature transform (SIFT) methods to match the detected face images with the trained face database by using the nearest distance.

2.1 LLE and ALLE for Dimension Reduction

For ease of the forthcoming discussion, the main features of the LLE algorithm is briefly introduced. It is an unsupervised learning algorithm that attempts to map high dimensional data to low dimensional, neighborhood preserved embeddings. It is based on the simple geometric intuitions: (i) each high dimensional data point and its neighbors lie on or are close to a locally linear patch of a manifold [7], and (ii) the local geometric characterization in original data space is unchanged in the output data space. From a mathematic point of view, the problem LLE attempts to resolve is: given a set $X = [x_1, x_2, \ldots, x_N]$, where $x_i (i = 1, \ldots, N)$ is i^{th} node on a high dimensional manifold embedded in R^D, i.e., $x_i \in R^D$, and then find a set $Y = [y_1, y_2, \ldots, y_N]$ in R^d, where $d \ll D$ such that the intrinsic structure in X can be represented by that of Y.

The main steps of LLE algorithm is described as follows:

Step 1: Neighborhood Selection
The K closest neighbors are selected for each point using a distance measure such as the Euclidean distance or be adaptively selected by our proposed method (will be mentioned later) to build ALLE.

Step 2: Optimal Weights Construction
The optimal weight matrix w_{ij} for data reconstruction can be obtained by minimizing the approximation error const function (reconstruction errors)

$$\epsilon(W) = \sum_i \left\| x_i - \sum_{j \in \Omega_i} (w_{ij} x_j) \right\|^2 \tag{1}$$

subject to the constraints

$$j \notin \Omega_i \Rightarrow w_{ij} = 0 \tag{2}$$

$$\sum_{j \in \Omega_i} w_{ij} = 1 \tag{3}$$

where $\boldsymbol{w}_i = [w_{i1}, \ldots, w_{ik}]$ are the weights connecting sample x_i to its neighbors.

Then, the optimal weights can be calculated through a least squares minimization of the reconstruction errors.

Step 3: Compute Low Dimensional Embedding
The final step of LLE is to compute a low dimensional embedding based on the reconstruction weights w_{ij} of the high dimensional inputs x_i. The low dimensional outputs $y_i,\ i = 1, 2, \ldots, N$ are found by minimizing the cost function representing locally linear reconstruction errors,

$$\Phi(Y) = \sum_i \left\| y_i - \sum_{j \in \Omega_i} w_{ij} y_j \right\|^2 \tag{4}$$

where $\boldsymbol{Y} = [\boldsymbol{y}_1, \ldots, \boldsymbol{y}_N]$ consist of the data points embedded into the low dimensional space. Zero mean and unity covariance is used in the LLE algorithm to make the minimization problem well-posed, that's to say, $\boldsymbol{Y}$ should obey the constraints

$$\sum_{i=1}^{N} y_i = \mathbf{0} \tag{5}$$

$$\frac{1}{N} \boldsymbol{Y} \boldsymbol{Y}^T = I \tag{6}$$

In [8], we have discussed that the the choice of K affects the tradeoff between the redundancy present in the structure and the number of isolated nodes. However, to manually

select K by trial and error is quite troublesome and time consuming. As a result, an adaptive scheme to select K is more appropriate for finding neighbors.

In this work, an adaptive neighborhood selection algorithm is used to build adaptive locally linear embedding (ALLE). The adaptive neighborhood selection algorithm is described in the following section.

2.2 Adaptive Neighborhood Selection

The effectiveness of manifold learning depends on the manner in which the nearby neighborhoods overlap with each other [9]. However, selecting the neighborhood size for face data set appropriately through trial and error procedure is very difficult because each face is a data point lying in a nonlinear embedded subspace. A possible solution is using adaptive locally linear embedding (ALLE) algorithm to do dimension reduction for the face samples.

ALLE is used to adaptively select the neighborhood size of each face sample. Essentially, the adaptive neighborhood selection algorithm is an iterative process of optimization. There are two parts in the algorithm, neighborhood contraction and neighborhood expansion. The detailed deduction of the algorithm can be find in [9].

For each sample point $\boldsymbol{x}_i$, we adaptively select k by a neighborhood contraction and expansion algorithm.

a. Neighborhood Contraction

Step 1: Empirically choose the initial neighborhood size K and $\boldsymbol{X}_i^k = [\boldsymbol{x}_{i1}, \ldots, \boldsymbol{x}_{ik}]$ is K nearest neighbors for data $\boldsymbol{x}_i$, ordered in nondecreasing distances to $\boldsymbol{x}_i$. Assign K to k.

Step 2: Subtract the mean value: $\boldsymbol{G} = \boldsymbol{X}_i^k - \bar{\boldsymbol{X}}_i^k$; compute the d largest singular vectors $\boldsymbol{Q}_i^k$ of $\boldsymbol{G}$ corresponding to d non-zero singular values and then project $\boldsymbol{G}$ to $\boldsymbol{Q}_i^k$: $\boldsymbol{E}_i^k = (\boldsymbol{Q}_i^k)\boldsymbol{G}$.

Step 3: Assume η is constant, if

$$||\boldsymbol{G} - \boldsymbol{Q}_i^k \boldsymbol{E}_i^k||_F < \eta ||\boldsymbol{E}_i^k||_F \tag{7}$$

then $\boldsymbol{X}_i = \boldsymbol{X}_i^k$, $\boldsymbol{E}_i = \boldsymbol{E}_i^k$, the neighborhood contraction is finished; otherwise, continue with the following steps.

Step 4: Assume $0 < k_0 < 1$, if $k > d + k_0$, delete the last column of $\boldsymbol{X}_i^k$ to get $\boldsymbol{X}_i^{k-1}$; set $k = k - 1$ and return to Step 2; otherwise go to Step 5.

Step 5: Let

$$k = \text{arcmin}_{d+k_0 \le j \le k} \frac{||\boldsymbol{X}_i^j - \bar{\boldsymbol{X}}_i^j - \boldsymbol{Q}_i^j \boldsymbol{E}_i^j||_F}{||\boldsymbol{E}_i^j||_F} \tag{8}$$

and set $\boldsymbol{X}_i = \boldsymbol{X}_i^k$, $\boldsymbol{E}_i = \boldsymbol{E}_i^k$.

Then $\boldsymbol{X}_i$ represents the contracted neighbors of $\boldsymbol{x}_i$. Step 4 and 5 mean that if there is no k-NN ($k \ge d + k_0$) satisfying (7), then the contracted neighborhood X_i should be one that minimizes $\frac{||\boldsymbol{X}_i^j - \bar{\boldsymbol{X}}_i^j - \boldsymbol{Q}_i^j \boldsymbol{E}_i^j||_F}{||\boldsymbol{E}_i^j||_F}$.

b. Neighborhood Expansion

Step 1: Set k to be the column number of X_i as the contracted neighborhood size obtained by the neighborhood contracting step, $\boldsymbol{x}_{ij}$ is the jth neighbor of $\boldsymbol{x}_i$, $\bar{\boldsymbol{x}}_i$ is the column mean of $\boldsymbol{X}_i$, $\boldsymbol{Q}_i$ represents the singular vectors. For $j = k+1, \ldots, K$, compute $\boldsymbol{\theta}_j^i = \boldsymbol{Q}_i^T(\boldsymbol{x}_i - \bar{\boldsymbol{x}}_i)$.

Step 2: if $||(\boldsymbol{I} - \boldsymbol{Q}_i\boldsymbol{Q}_i^T)(\boldsymbol{x}_{ij} - \bar{\boldsymbol{x}}_i)||_2 \leq ||\boldsymbol{\theta}_j^i||_2$, then expand $\boldsymbol{X}_i$ by adding $\boldsymbol{x}_{ij}$.

In summary, the above adaptive neighborhood selection algorithm is validated by minimizing the ratio as objective function:

$$\Delta = \frac{||\boldsymbol{X}_i - \boldsymbol{X}_i - \boldsymbol{Q}_i\boldsymbol{E}_i||_F}{||\boldsymbol{E}_i||_F} \tag{9}$$

Given this adaptive neighbor selection algorithm, we can then utilize the adaptive neighbor selection criterion to enhance the LLE algorithm and form the adaptive locally linear embedding (ALLE).

In the following sections, the above dimension reduction algorithm LLE and ALLE will be applied to SIFT algorithm to form new method for face feature extraction, called LLE/ALLE-SIFT.

2.3 Scale-Invariant Feature Transform (SIFT)

SIFT is an algorithm popularly used in computer vision to detect and describe local features in images, whose applications include object recognition, robotic mapping and navigation, image stitching, 3D modeling, gesture recognition and video tracking [10, 11, 12, 13].

SIFT features are local information based on the appearance of the object at particular interest points. They are not only invariant to image scale and rotation, but also robust to the changes of illumination, noise and viewpoint. In addition to these properties, they are highly distinctive, relatively easy to extract, and allow for correct object identification with low probability of mismatch, which lead to easy and convenient way for matching against a large database of local features. Object description by a set of SIFT features is also robust to partial occlusion. It is noted that just 3 SIFT features from an object are enough for computing its location and pose. The four major stages of SIFT are described in [13]. For concision, they will not be repeated in this paper.

The construction of the standard SIFT feature vector is complicated and the choices behind its specific design as described in [13] are not clear. In [14], a more distinctive representation for local image descriptors is proposed and proved by experiment to be theoretically simpler, more compact, faster and more accurate than the standard SIFT descriptor, known as PCA-SIFT. As mentioned above, the goal of keypoint descriptor is to create a descriptor for the patch that is compact, highly distinctive and robust to changes. The idea of PCA-SIFT is using PCA to reduce the dimensionality of the 128-dimensional feature vector to obtain a more compact and distinctive descriptor. Based on the consideration, we consider utilizing another famous dimension reduction algorithm, locally linear embedding (LLE), and its extension adaptive locally linear embedding (ALLE) to do the dimension reduction of the feature vector to obtain a more compact and robust keypoint representor. This method is named as LLE/ALLE-SIFT.

The features constructed by SIFT, PCA-SIFT, LLE/ALLE-SIFT are then used for face recognition with some matching strategies as described in the following section.

2.4 Matching Strategies

As mentioned above, all the features of the training face images are extracted by LLE/ALLE-SIFT, and then saved into predefined database. The database is predefined as feature table which consists of a name field, a 1×4 vector (including (x, y) position, scale and orientation) and a 1×128 vector descriptor as shown in Fig. 1.

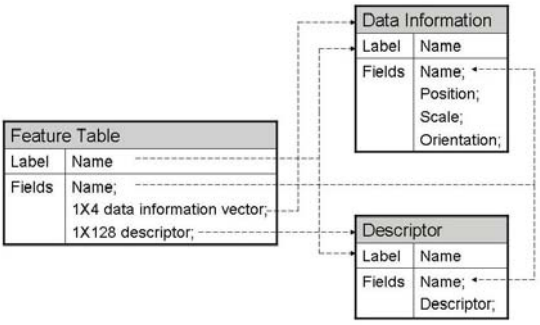


Fig. 1. Database description

After the feature tables of training face images are stored, SIFT is applied to the testing image to obtain the testing face SIFT features. Then the testing face descriptor vector is compared with the training face descriptor vectors by similarity using Euclidean distance measurement. If the Euclidean distance between the key point of a testing face descriptor and the key point of a training face descriptor falls below a chosen threshold, this pair is termed a match. Then, which class this testing image belongs to can be determined by labeling it with the same label of the training face image with maximum matching numbers.

3 Experiments and Results

To evaluate the performance of the proposed face recognition methods by experiments, the evaluation metrics used to quantify the experiment results are needed. In following sections, evaluation metrics is chosen and the outline of the experimental setup is discussed. Finally, the experiment results are displayed.

3.1 Experiment Setup

We have constructed the vision system for the robot which makes use of a Canon VC-C3 PTZ camera, interfaced with a Matrox Imaging/Meteor II frame grabber. Programming

is performed with Microsoft Visual Studio .NET 2003 in the C++ language, supported by open source programs obtained from OpenCV.

The database is constructed in a way to support recognition of individuals from the frontal view to quarter-profile view, not exceeding a rotation of more than 45 degrees of the head in the left or right direction. The recognition database contains the facial images of 6 different individuals, with 12 different training images for each individual. 6 images are frontal shots of the individuals with minor differences in orientations, while the other 6 images are quarter-profile shots in the left and right directions. A maximum limit of 12 testing images are set for each individual.

This face dataset is used to evaluate the algorithms and compare the difference between the standard SIFT representation and LLE/ALLE-SIFT. Three descriptors were evaluated in these experiments: (i) the standard SIFT feature representation (denoted as "SIFT"); (ii) PCA-SIFT as described in [14] (denoted as "PCA-SIFT", and we empirically determined good values for the dimensionality of the feature space, $n = 20$); (iii) LLE-SIFT as described above (denoted as "LLE-SIFT", and the neighborhood size is determined as $k = 24$, the feature space dimensionality $n = 20$); (iv) ALLE-SIFT which use ALLE instead of LLE (denoted as "ALLE-SIFT", which have a initial neighborhood size $K = 24$, and a feature space dimensionality of $n = 20$).

3.2 Evaluation Metrics

In this study, recall and 1-precision are chosen to do the evaluation since face recognition problem is a detection rather than a classification problem. The evaluating process is described as follows.

The keypoints for all of the images in the training dataset are identified using the initial stages of the SIFT algorithm. When a new testing image come, the key points of it are extracted and matched with that of the training images using the matching strategy mentioned above. If the Euclidean distance between the feature vectors for a particular pair of keypoints falls below the chosen threshold, this pair is termed a match. We compare the match numbers of pairs of the testing image and each training image, which is obtain from 6 different individuals, and each individual has stored 12 training images. If the match number of a pair of images exceed a predefined threshold, N, this match is called a positive one, otherwise it is negative. The label of the training data who have the most positive match to the testing image will become the label of the new testing image.

For easy presenting, a true-positive is a positive match between a testing image and a training image corresponding to the same subject; a false-positive is a positive match where the testing image and training image are from different subject; a false-negative is a negative match where the testing image and training image are from the same subject; a true-negative is a negative match where the testing image and training image are from different subject. We define N_{tp} as the number of true-positives, N_{fp} as the number of false-positives, N_{tn} as the number of true-negative, N_{fn} as the number of false-negative.

From statistical point of view, the Precision for a class is the number of true positives (i.e. the number of items correctly labeled as belonging to the class) divided by the total number of elements labeled as belonging to the class (i.e. the sum of true positives and

false positives, which are items incorrectly labeled as belonging to the class). Recall in this context is defined as the number of true positives divided by the total number of elements that actually belong to the class (i.e. the sum of true positives and false negatives, which are items which were not labeled as belonging to that class but should have been). The recall and precision can be expressed as:

$$\text{recall} = \frac{N_{tp}}{N_{tp} + N_{fp}} \tag{10}$$

$$\text{precision} = \frac{N_{tp}}{N_{tp} + N_{fn}} \tag{11}$$

In addition, the accuracy rate can be defined as:

$$\text{accuracy} = \frac{N_{tp} + N_{tn}}{N_{tp} + N_{tn} + N_{fp} + N_{fn}} \tag{12}$$

3.3 Results

In this section, face recognition experiment results are presented comparing LLE/ALLE-SIFT to the standard SIFT representation on the face images experiment. The training face dataset contains face images of 6 persons, each of which has 12 different pictures taken from different camera viewpoints.

Table 1. Specification of Face Datasets

Name	No. of Observations		Gender	Nationality
	Training data	Testing data		
Subject 1	12	8	Male	India
Subject 2	12	10	Male	Singapore
Subject 3	12	11	Male	Singapore
Subject 4	12	12	Male	Iran
Subject 5	12	8	Female	Iran
Subject 6	12	8	Female	China

Table 2. Performance Comparison of Different Feature Descriptors (N=5)

Name	SIFT			PCA-SIFT		
	Recall (%)	Precision (%)	Accuracy (%)	Recall (%)	Precision (%)	Accuracy (%)
Subject 1	53.8	63.6	84.8	96.3	97.4	98.3
Subject 2	83.3	90.9	95.5	71.6	74.9	81.9
Subject 3	83.3	90.9	95.5	78.9	87.8	90.2
Subject 4	72.7	72.7	90.9	82.9	90.0	94.8
Subject 5	100	75	96.9	69.0	69.9	78.8
Subject 6	84.6	100	96.9	69.6	78.4	89.5
Total	78.8	77.6	93.4	78.1	83.1	88.9

Table 3. Performance Comparison of Different Feature Descriptors (N=5)

Name	LLE-SIFT			ALLE-SIFT		
	Recall (%)	Precision (%)	Accuracy (%)	Recall (%)	Precision (%)	Accuracy (%)
Subject 1	83.8	78.6	87.2	96.3	97.1	99.2
Subject 2	73.3	80.9	90.5	71.6	94.9	91.9
Subject 3	73.3	75.9	86.5	78.9	97.8	88.2
Subject 4	82.7	79.7	90.9	82.9	97.0	94.6
Subject 5	78.2	60.5	84.3	69.0	89.9	88.9
Subject 6	64.9	72	83.4	69.6	98.3	94
Total	72.7	72.7	87.1	84.6	96.5	92.8

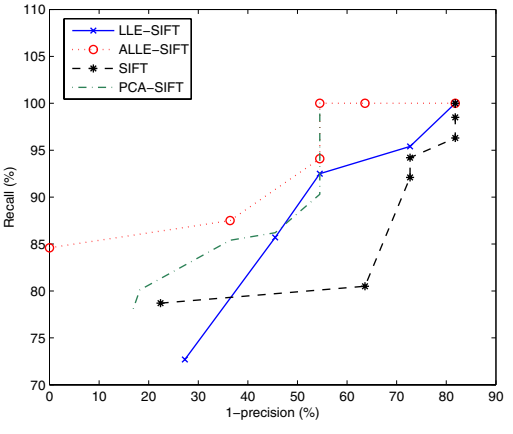


Fig. 2. Recall and 1-precision: SIFT vs. PCA-SIFT vs. LLE-SIFT vs. ALLE-SIFT

Table 1 shows the statistic characteristic of the dataset. The goal was to match corresponding keypoints between images to determine which class the newly coming testing images belong to. The experiment results are shown in Tables 2- 3.

The recall vs. 1-precision graphs are generated for experiments by varying the threshold for each algorithm. Figure 2 shows the matching performance for the four algorithms on the described dataset. From the figure, PCA-SIFT, LLE-SIFT and ALLE-SIFT clearly dominate. Especially, ALLE-SIFT obtain better recall at high precision.

4 Conclusions

In this paper, an intelligent vision human detection system for identifying human through face recognition was described. Feature extraction algorithms, SIFT and LLE/ALLE were combined to form new method for finding compact and robust feature descriptor to do face recognition. The proposed robot vision system for human recognition was tested in experiments of our face datasets.

Acknowledgement

This study was partially supported by Singapore National Research Foundation, Interactive Digital Media R&D Program, under research grant R-705-000-017-279 and National University of Singapore, under URC Funding: Mind Robotics (R263000490112).

References

1. Zhang, C.S., Wang, J., Zhao, N.Y., Zhang, D.: Reconstruction and analysis of multi-pose face images based on nonlinear dimensionality reduction. Pattern Recognition 37(2), 325–336 (2004)
2. Lee, C.H., Kim, J.S., Park, K.H.: Automatic human face location in a complex background using motion and color information. Pattern Recognition 29(11), 1877–1889 (1996)
3. Dai, Y., Nakano, Y.: Face-texture model based on sgld and its application in face detection in a color scene. Pattern Recognition 29(6), 1007–1017 (1996)
4. Scholkopf, B., Smola, A., Muller, K.R.: Nonlinear component analysis as a kernel eigenvalue problem. Neural Computation 10(5), 1299–1319 (1998)
5. Lu, J., Plataniotis, K.N., Venetsanopoulos, A.N.: Face recognition using kernel direct discriminant analysis algorithms. IEEE Transactions on Neural Networks 14(1), 117–126 (2003)
6. Yang, J., Frangi, A.F., Yang, J.-y., Zhang, D., Jin, Z.: KPCA plus LDA: A complete kernel fisher discriminant framework for feature extraction and recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence 27(2), 230–244 (2005)
7. Roweis, S.T., Saul, L.K.: Nonlinear dimensionality reduction by locally linear embedding. Science 290, 2323–2326 (2000)
8. Pan, Y., Ge, S.S., Mamun, A.A.: Weighted locally linear embedding for dimension reduction. Pattern Recognitioin 42, 798–811 (2009)
9. Wang, J., Zhang, Z., Zha, H.: Adaptive manifold learning. In: Saul, L.K., Weiss, Y., Bottou, L. (eds.) Advances in Neural Information Processing Systems 17, pp. 1473–1480. MIT Press, Cambridge (2005)
10. Lowe, D.G.: Object recognition from local scale-invariant features. In: Proceeding of International Conference on Computer Vision, Corfu, Greece, September 20-27, pp. 1150–1157 (1999)
11. Lowe, D.G.: Local feature view clustering for 3D object recognition. In: Proceeding of IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2001), Kauai Marriott, Hawaii, US, pp. 682–688. Springer, Heidelberg (2001)
12. Brown, M., Lowe, D.G.: Invariant features from interest point groups. In: Proceeding of British Machine Vision Conference, Cardiff, Wales, September 2-5, pp. 656–665 (2002)
13. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision 60(2), 91–110 (2004)
14. Ke, Y., Sukthankar, R.: PCA-SIFT: A more distinctive representation for local image descriptors. In: Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), Washington DC, USA, June 27 - July 2, pp. 506–513 (2004)

The Influence of Social Interaction on the Perception of Emotional Expression: A Case Study with a Robot Head

John C. Murray[1],Lola Cañamero[1], Kim A. Bard[2], Marina Davila Ross[2], and Kate Thorsteinsson[2]

[1] Adaptive Systems Research Group, University of Hertfordshire College Lane, Hatfield, AL10 9AB, UK
[2] Centre for the Study of Emotion, Department of Psychology, University of Portsmouth, PO22 0JN, UK

Abstract. In this paper we focus primarily on the influence that socio-emotional interaction has on the perception of emotional expression by a robot. We also investigate and discuss the importance of emotion expression in socially interactive situations involving human robot interaction (HRI), and show the importance of utilising emotion expression when dealing with interactive robots, that are to learn and develop in socially situated environments. We discuss early expressional development and the function of emotion in communication in humans and how this can improve HRI communications. Finally we provide experimental results showing how emotion-rich interaction via emotion expression can affect the HRI process by providing additional information.

Keywords: Robotics, HRI, Emotion Expression, Social Interaction.

1 Introduction

This paper presents work done under the European Commission funded project FEELIX-GROWING. The project aims to develop robotic architectures and algorithms that allow for robots to learn and develop in socially situated environments, i.e. allowing the robot to take direct input from the world around it, growing and learning over time as opposed to taking a fully 'developed' off-the-shelf robot and therefore allowing for robots that respond uniquely to individual people. This is a developmental process that resembles that of early emotional communication in infants.

In developing robots to be integrated into human environments and work and interact closely with humans (such as service and rehabilitation robots) it is important that the HRI process be as 'natural' as possible, i.e. allowing for the human to interact and understand the robots actions without (or with very little) prerequisite knowledge of the system, indeed communicating as one would intuitively with another human.

The aim of this paper is to show the use of expression in communication and more specifically the effect interaction has on the perception and recognition of

J.-H. Kim et al. (Eds.): FIRA 2009, LNCS 5744, pp. 63–72, 2009.

emotion expressions. Section 2 provides information on human facial expression, its uses in communication and advantages that it can provide in interactions. Section 3 discusses the need for emotion expression in socially situated robots, and how this can theoretically improve feedback from the robot to the human participant. Section 4 describes the interaction scenarios and experimentation whilst section 6 draws the conclusions from the results of the experiments, and finally section 6.1 describes the further work to be carried out.

2 Human Facial Expression

2.1 Early Expressional Development

Every-day experience shows us that facial expressions can so evidently convey emotions, a trait that led to a series of systematic approaches to study emotions based on facial muscle activations [3], [15]. Similar to other modes of emotional expressions, facial expressions may not only cause valence-, arousal- and knowledge-based changes and behavioural responses in individuals emitting them, but they may also notably affect the inner and outer states of their recipients [6], which themselves may trigger further exchanges among social partners as part of the ongoing dynamics of social communication.

Some facial expressions are thought to be innate, as they are present in human newborns, as well as in newborn chimpanzees, our close evolutionary cousins (e.g., endogenous smiles [10], cry face [1]). However, it is now thought very early in life, facial expressions develop both communicative and expressive functions as a function of, and in response to socio-emotional interactions with others. As a result, emotional expressions change from those innate forms that are present at birth. With development across the first months, some new expressions emerge, and some expressions become more complex [4]. These include graded expressions of distress (lip pouting, chin quivering accompanied by fussy cries, to the typical cry face, usually accompanied by vocal crying [1], [14], [4]), expressions of disgust [20], and even smiling [14]. It is interesting to note that anger and fear expressions also emerge by 4-6 months in human infants, becoming differentiated from the more general distress expressions of newborns [4].

Emotional development is a process that involves infant responsitivity to caregivers, in interaction with caregivers responsitivity to infants [4], [21]. Social processes, some of which are culturally diverse [8], contribute to the development of functionality in early emotional expressions. For instance, the responsive smiles of a 3-month-old infant function to enhance mothers sense of emotional engagement. Caregivers respond contingently to positive infant expressions, and infants learn to take turns in behavioural proto-conversations [21]. By 4-6 months, infants are active participants in elaborated social games, e.g. peek-a-boo [5].

The peek-a-boo paradigm, a naturalistic face-to-face interaction that involves hidden and revealed emotional expressions embedded within a positive social interaction, was developed as part of the FEELIX-GROWING project, to investigate the positive emotional and social engagements of 4-month-old human infants with adults. Thus, the peek-a-boo paradigm may well prove to be useful

to enhance the emotional engagement of humans and interactive social robots. Therefore, as we discuss further in later sections, this is the reason for the peek-a-boo scenario being used in the study presented in this paper.

2.2 Use in Communication

Facial expression is used by all typically developed humans when engaged in communication processes [12] "The function of emotions is communicative: Emotions communicate both to ourselves and others". Oatley also proposes that one of the possible reasons for subconscious emotion expression is due to the cognitive system being a multi-layered structure where we as individuals do not have direct control of the lower levels. However, they propose that the top most level, receiving its input from the lower levels, has some level of awareness of the models goals, [7].

There are many different expressions that can be displayed by a human face with varying levels of intensity, e.g. from a gentle smile to a wide open mouth or play smiles. Each facial expression can be identified by the individual movements of the facial muscles. One tool that enables detailed coding of facial expression is the Facial Action Coding System (FACS) [3]. This system enables trained observers to identify and record individual facial muscle movements, each muscle action denoted by a specific Action Unit (AU).

For the purposes of the robotic system presented in this paper, our goal is to create a robot that is capable of presenting emotional expressions as additional sensory input to participants to enable better understanding and perception of the intent of the robots actions.

3 Emotional Expression

Emotion expression is also an important aspect of human communication, and provides us with many ways to enrich the interaction in addition to providing important information and cues. Emotional expressions are driven by the internal emotional state of the participant. Keeping in context with the goals of the larger project FEELIX-Growing for which this research has been carried out; it can be seen that 87% of mothers notice a strong emotional expression, such as 'anger', from infants as young as three months [18].

It is presumed in [9] that emotional expression in infants is a direct correlation to a representation of an underlying emotional state. Lewis et al [9] have shown that during a simple learning task, infants move through a range of emotional expressions that appear appropriate for the current phase of learning. Therefore, based on these findings it is safe to assume that emotion expression is a communicative system that exists from a very early stage of development, allowing for non-verbal communication of emotion, mood, and to some extent, intent, regarding that current task, that is innate in all humans and presumably understood.

3.1 Emotion Expression in HRI

There exists many different types of socially interactive robots capable of communicating in a variety of ways, from arm gesture movement, and body posture movement [16], to touch [2], in addition to other types of interface. These allow for effective interactive robots in social situations. However, in order to interact with these systems effectively, prerequisite knowledge is usually required, knowing how to interpret the interface on the robot or what responses/instructions to give the robot. Other interactive gesture robots such as BERTI require that any social interaction from a human participant be conducted whilst fitted with an IR Glove so that the system can monitor the humans movements.

As previously discussed, emotion expression is an important aspect of two-way communication and has been seen in infants as young as three months. Therefore we have tested two interactive scenarios designed to use the emotion expressions available with our robot.

3.2 ERWIN - Socially Interactive Robot

ERWIN is a multi-modal robotic head that can interact visually or acoustically with a participant. It has the ability to detect faces in the scene, extracting and learning facial feature metrics to later be able to recognise the same person [11]. ERWIN also has the capability to recognise different interactants based on their voice, learning features of a persons voice as they interact with the robot, therefore allowing the robot to attend just to a specific person, for example the caregiver. ERWIN has six degrees of freedom, two for movement of the robot in pan and tilt, with the remaining four used to control the emotion expression, i.e. two for eyebrows, and two for lips. With these degrees of freedom, it is possible to express five emotions, these are shown in fig. 1 and include: happy, sad, surprised, afraid and angry.

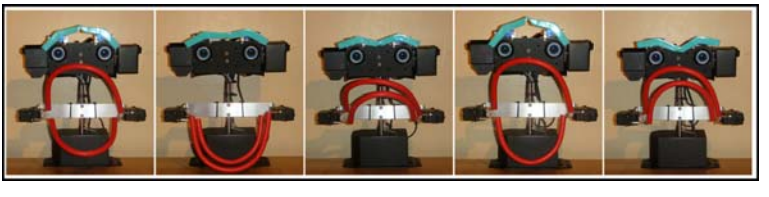


Fig. 1. The range of emotion expressions available with ERWIN, from left to right - Afraid, happy, sad, surprised and angry

4 Experimentation Scenarios

The main hypothesis behind the experimentation provided in this paper was to test if particular affective interactions with the robot could improve a participants ability to interpret what emotion expression is being expressed by the robot, and providing an improvement in the overall interaction process.

In order to test this hypothesis, two separate interaction scenarios were devised. In scenario 1 the affective goal was for ERWIN to attempt to engage and attract the attention of the participant via emotion expression, person tracking and 'fidgeting' movements (see section 4.1). For scenario 2, the participants are asked to play a game of peek-a-boo with ERWIN (see section 4.2). The aim in scenario 2 was for the participant to attract the attention of ERWIN and engage the robot. Each scenario lasted for 1 minute, with the full range of ERWIN's emotion expressions being exhibited during the interactions.

For the two scenarios, we had twenty participants take part with an age range of 21 to 53 which included 11 males and 9 females. During the experimentation the order of the two scenarios was alternated. This is to ensure that the order of the scenarios did not have an effect on the outcome of the experiments.

4.1 Scenario 1

As previously mentioned, scenario 1 was designed to allow the robot to try to initiate interaction with the participant. The robot had been left for sometime without interaction and had therefore expressed a particular emotional expression, in this case 'sad'. The participant was instructed to then sit in the chair in front of the robot and engage very little with the robot, mainly ignoring it.

This scenario allowed for the full range of ERWINs emotional expressions to be shown, from the following interaction cues; when no person was detected by the robot over a long period of time the 'sad' expression was exhibited. Once a participant began the scenario, 'surprise' was expressed. This provided an instant visual cue for the participant to relate the action to their presence. If the participant looked at the robot 'happy' was expressed and if they looked away from the robot, choosing to ignore it, 'sad' would be expressed followed, after a some time, by 'angry'. Figure 2 shows a participant looking away from ERWIN, paying no attention and choosing not to engage, ERWIN expresses 'anger' due to the duration the participant has ignored it. The final expression to be used in this scenario is 'afraid'. When there was no engagement with ERWIN, it would alter its expression, but also begin looking around the environment, moving (sometimes randomly), trying to gain the attention of the participant so as to engage. The participants were told that they could, at their discretion, tell the robot to 'STOP!' as to treat it as being too persistent, hence expressing 'fear'.

4.2 Scenario 2

The affective component of the second scenario was the opposite of that of scenario 1. Scenario 2 required that the participant try and engage with the robot, this was achieved by playing peek-a-boo. As previously discussed, Peek-a-boo is a simple game based on face-to-face interaction, commonly played to elicit positive affect in infants. It is a naturally occurring face-to-face event that incorporates aspects of attention, mutual engagement, emotional expression, social expectation and importantly, the mutual goal of sharing positive emotional

Fig. 2. Scenario 1 - ERWIN fails to gain the participants attention, therefore expressing anger after repeated failed attempts

exchange. By the age of 4 months, typically developing infants are skilled in mutual gaze with mother and inter subjectivity, the sharing of affect and intention in a turn-taking, timing-sensitive, contingent interaction.

Therefore, we chose the game of peek-a-boo as the basis of scenario 2 for several reasons; peek-a-boo being a common infant's game, requiring very little verbal communication. In developmental psychology peek-a-boo is also used to test social expectations and emotions in infants [19] therefore making it ideal to use for the testing of socially situated emotion in HRI. The simplicity of peek-a-boo, the basic premise being one hides their face from view using ones hands or hiding behind an object and then reappearing, making it ideal to implement in a robotic system [13].

Figure 3 shows a participant playing peek-a-boo with ERWIN. In the left side of the figure the participant is hiding their face, therefore ERWIN changed his expression, however, in the right side the participant 'reappears' and this is reflected in ERWINs change of emotion.

Fig. 3. Scenario 2 - The human plays Peek-a-boo with the robot by hiding their face, then tried to engage the robot by removing their hands

5 Results

During the experimentation the participants were asked to complete a questionnaire in three stages, before any robot interaction, after Scenario 1 and finally after Scenario 2.

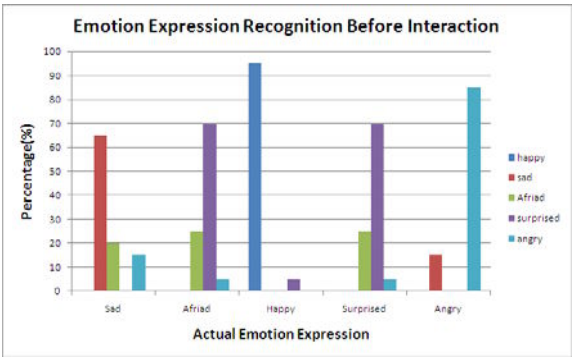


Fig. 4. Results of expression identification before interaction

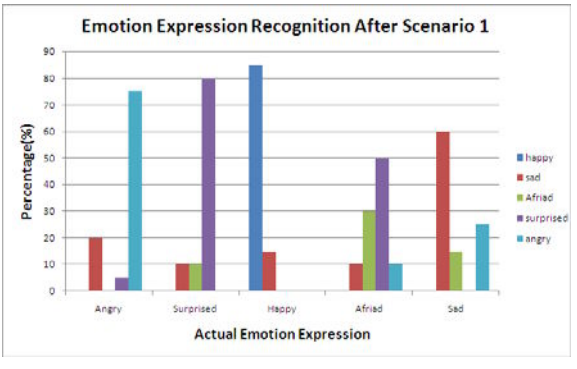


Fig. 5. Results of expression identification after scenario 1

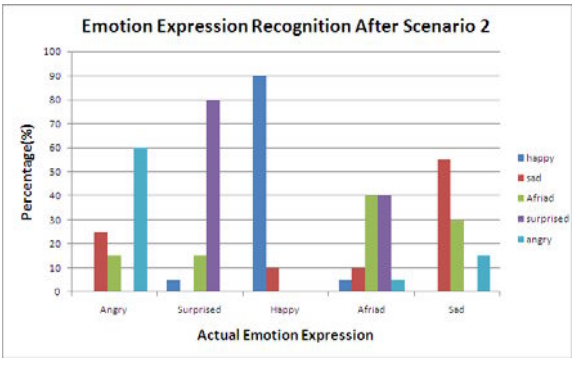


Fig. 6. Results of expression identification after scenario 2

Before the participants interacted with ERWIN, they were given an initial set of images representing ERWINs emotion expressions and were asked to identify what they thought each image represented. The options available were: happy, sad, angry, surprised and afraid. After the participants had engaged with the robot in the first scenario they were again given the same set of images and asked to identify the emotion expressions (using the same set of answers). Finally, after the second interaction scenario, the participants are given the images one final time to again select what emotion expression is exhibited.

Figures 4, 5 and 6 show the results of the expression identification tasks by the participants. As can be seen in fig 4 over 65% of the participants correctly identified the emotion expressions, with the most accurate being 'happy' (see fig 1) identified correctly by 95% of participants. An expression was deemed 'correct' when the participant matched the expression identified by the designers. The emotional expressions of ERWIN were designed intuitively. See 6.1 for more details.

After the first interaction scenario, the results of the participants emotion expression selections are shown in fig 5. As can be seen from this graph, the identification results have changed somewhat with the biggest change being in 'afraid' falling by 20%. However, from this, what is interesting to see, is with 'afraid' and 'angry', whilst the recognition rates by the participants have fallen, the distribution of the categorisation of the emotions has widened.

After the second interaction scenario, again the participants made their selections about the emotion expressions and the results of which are shown in fig 6. Shown here is that after the second interaction some of the participants again selected different responses to their previous choices. After scenario 2, 15% of people selected 'afraid' for the 'angry' selection, with 30%, up from 15% after scenario 1, selecting 'afraid' for the 'sad' expression.

These results show that, whilst there is some misrepresentation of the emotion expressions within the images, initially the majority of participants are able to identify the correct facial expressions. However, it can be clearly seen from the results in figs 4-6 that the interaction scenarios have had a noticeable affect on the emotion identification classification, whether positively or negatively.

When asked which interaction scenario the participants preferred 60% chose Scenario 2, the main decision for this was due to the robot appearing to enjoy the interaction more, in addition to their 'being more [for the participant] to do' in the scenario.

6 Conclusions

It was very important to ensure that interactive scenarios were chosen that elicited an opposite affective reaction from the participants. It was for this reason the two complementary scenarios described in sections 4.1 and 4.2 were chosen. With the first requiring the robot to engage whilst the second requires the human participant to assume the role of gaining the robots attention for engagement.

It is apparent from the results in section 5 that affective social interaction involving two way communication can affect the perception of emotion expression

in HRI. We have shown how a game of peek-a-boo can elicit a change in perception of the emotional expressions of a robot. Whilst at present we have only noticed a 15% change in perception, this allows us to ask further questions for the development of HRI scenarios, such as: how does more emotional expressivity effect: peek-a-boo performance, the interactants facial expression and perception of the social situation/interaction, or the quality of the interaction. Additionally we can probe how the modality of interactions (touch, vision, vocalisations) will affect this perception (see section 6.1).

6.1 Future Works

The conclusions drawn from this work are as follows, there are two important changes to be made to the scenarios and design of the system for future interactions and experiments. It is clear that the visual information provided via the emotion expressions displayed by ERWIN during interaction has an effect on the responses given by the participants. In more detailed explanations on their interpretations, it is clear that some of the participants were not seeing the lower section of the robot as the mouth, but more as hands or some other extension of the robot. This created some misunderstanding as to what emotion was being represented. Therefore, future work will examine modifying the emotion expressions to conform more accurately to FACS movements.

Acoustics can provide additional information when communicating in an interactive process, not just with spoken words but non-verbal vocalisations. This is common in infants and babies at the 'Pre-Intentional: Reflexive Level', this discusses how an infant is limited to simple reflexes and basic vocalisations such as cries, vegetative noises, and laughing, not to mention facial expressions [17]. Therefore, we hope that integrating this in the near future will dramatically improve the social interaction.

Acknowledgements

We would like to thank Nicolas Oros for helping with the experimentation. Work done as part of FEELIX GROWING www.feelix-growing.org.

References

1. Bard, K.A.: Emotions in chimpanzee infants: the value of a comparative developmental approach to understand the evolutionary bases of emotion. In: Nadel, J., Muir, E.(eds.) Emotional Development: Recent Research Advances, pp. 31–60. Oxford University press, Oxford (2005)
2. Cañamero, L.D., Fredslund, J.: I show You How I like You-Can You Read it in My Face? IEEE Transactions on Systems, Man and Cybernetics, Part A 31(5), 454–459 (2001)
3. Ekman, P., Friesen, W.V.: Facial action coding system: Investigator's guide. Consulting Psychologists Press, Palo Atto (1978)

4. Fogel, A.: Infancy: Infant, Family, and Society, 4th edn. Wadsworth, Belmont (2001)
5. Greenfield, P.M.: Playing Peekaboo with a 4-Month-Old - Study of Role of Speech and Nonspeech Sounds in Formation of a Visual Schema. Journal of Psychology 82(2), 287–292 (1972)
6. Hatfield, E., Cacioppo, J.T., Rapson, R.L.: Emotional Contagion. Cambridge University Press, Cambridge (1994)
7. Johnson-Laird, P.N.: A computational analysis of consciousness. Cognition and Brain Theory 6, 499–508 (1983)
8. Keller, H.: Cultures of Infancy. Erlbaum, Mahwah (2007)
9. Lewis, M., Alessandri, S., Sullivan, M.W.: Violation of expectancy, loss of control, and anger in young infants. Developmental Psychology 26, 745–751 (1990)
10. Mizuno, Y., Takeshita, H., Matsuzawa, T.: Behavior of Infant Chimpanzees during the night in the first 4 months of life: Smiling and suckling in relation to behavioural state. Infancy 9(2), 221–240 (2006)
11. Murray, J.C., Wermter, S., Knowles, M.: MIRA: A Learning Interactive Robot Agent. In: Proceedings of the IEEE Conference on Hybrid Intelligent Systems, HIS 2008, Barcelona, Spain (2008)
12. Oatley, K., Manstead, A.S.R.: Best Laid Schemes: The Psychology of the Emotions - Studies in Emotion and Social Interaction, pp. 44–60. Cambridge University Press, Cambridge (1992)
13. Ogino, M., Ooide, T., Watanabe, A., Asada, M.: Acquiring Peekaboo Communication: Early Communication Model Based on Reward Prediction. In: IEEE 6th International Conference on Development and Learning, pp. 116–121 (2007)
14. Oster, H.: The repertoire of infant facial expressions: An ontogenetic perspective. In: Nadel, J., Muir, D. (eds.) Emotional Development. Oxford University Press, Oxford
15. Oster, H.: Emotion in the Infant's face. In: Annals of the New York Academy of Sciences, vol. 1000, Issue Emotions inside out: 130 Years after Darwin's The Expression of the Emotions in Man and Animals, pp. 197–204 (2006)
16. Otero, N.R., Nehaniv, C.L., Syrdal, D.S., Dautenhahn, K.: Naturally Occurring Gestures in a Human-Robot Teaching Scenario, Interaction Studies (In Press) (2008)
17. Owens, R.E.: Language Development: An introduction, 3rd edn. Merrill, New York (2001)
18. Pannabecker, B.J., Emde, R.N., Johnson, W., Stenberg, C., David, M.: Maternal Perceptions of infant emotions from birth to 18 months: A preliminary report. In: Paper presented at the International Conference of Infant Studies, New Haven, CT (1980)
19. Rochat, P., Querido, J.G., Striano, T.: Emerging Sensitivity to the Timing and Structure of Protoconversation in Early Infancy. Developmental Psychology 35(4), 950–957 (1999)
20. Steiner, J.E., Glaser, D., Hawilo, M.E., Berridge, K.: Comparative Expression of Hedonic Impact: Affective Reactions to Taste by Human Infants and Other Primates. Neuroscience and Biobehavioral Reviews 25, 53–74 (2001)
21. Trevarthen, C., Aitken, K.J.: Infant intersubjectivity: Research, theory, and clinical applications. Journal of Child Psychology and Psychiatry and Allied Disciplines 42(1), 3–48 (2001)

Towards Humanlike Social Touch for Prosthetics and Sociable Robotics: Handshake Experiments and Finger Phalange Indentations

John-John Cabibihan[1,2], Raditya Pradipta[2], Yun Zhi Chew[2], and Shuzhi Sam Ge[1,2]

[1] Social Robotics Laboratory, Interactive and Digital Media Institute and
[2] Department of Electrical and Computer Engineering,
National University of Singapore, 4 Engineering Drive 3, Singapore 117576
{elecjj,raditya.pradipta,samge}@nus.edu.sg, kane.cyz@gmail.com

Abstract. The handshake has become the most acceptable gesture of greeting in many cultures. Replicating the softness of the human hand can contribute to the improvement of the emotional healing process of people who have lost their hands by enabling the concealment of prosthetic hand usage during handshake interactions. Likewise, sociable robots of the future will exchange greetings with humans. The soft humanlike hands during handshakes would be able to address the safety and acceptance issues of robotic hands. This paper investigates the areas of contact during handshake interactions. After the areas of high contact were known, indentation experiments were conducted to obtain the benchmark data for duplication with synthetic skins.

Keywords: Social Robotics, Prosthetics, Rehabilitation Robotics, Human Handshake.

1 Introduction

From birth to old age, one of the basic needs of humans is to touch and be touched. Many researchers have long acknowledged the positive benefits of the human touch. In nursing care, it is known that the human touch promotes the physical, emotional, social and spiritual comfort [1]. Touch was also shown to be an effective means to persuade someone and to gain compliance; a brief touch on the upper arm, for example, increases people's altruistic behavior and willingness to comply to a request [2, 3]. In social interactions, it is common that touches are exchanged – a handshake, a hug, high fives. Among these, the handshake emerged to be the primary gesture to exchange greetings in many cultures. Hence, the objective of this paper is to conduct handshake experiments and determine the areas of high contact. Once these areas are known, the human hand is to be indented to obtain force-displacement data. The accompanying paper demonstrates how synthetic skins could then be designed according to the biomechanical behavior of the skin of the human hand during handshake interactions [4].

J.-H. Kim et al. (Eds.): FIRA 2009, LNCS 5744, pp. 73–79, 2009.

For prosthetic hands, the possibility to duplicate the softness of the human hand for the synthetic skins could have significant benefits on the well-being of a person who lost his or her hand, which may be due to accidents, disease or anomalies at birth. The primary requirement of prosthetic devices is to allow a user to pass unnoticed [5]. The availability of artificial hands with humanlike appearance but not humanlike softness poses a perception problem to the person that these hands will touch. As such, finding synthetic materials that have similar properties to that of human skin becomes important [6]. This artificial skin material has to exhibit not only similar appearance, but also similar softness. Softness has been defined as the perceptual correlate of skin compliance, which is the amount of deformation caused by an applied force [7]. Moreover, it can be expected that in the future, sociable robots will be able to greet humans in a similar fashion. The hand will be the area where contact with human will occur most frequently. Whereas in most of the previous papers the focus was in the motion of the handshake, this paper will focus on the exerted force during handshake process. This paper is organized as follows. First, the handshake experiments are described and the contact forces during handshakes are presented. Next, the areas of the hand where the high contact areas occurred are indented and the force-displacement curves obtained. The paper ends with the concluding remarks.

2 Handshake Experiments

The objectives of the handshake experiments are to determine: (i) the locations of the high contact areas and, (ii) the contact forces involved. For the purposes of designing synthetic skins for handshake interactions, it would be meaningful to focus the design on the areas of high contact under the typical forces during handshaking.

2.1 Methodology

It was earlier observed in Ref. [8] that men tend to shake hands while women embrace. In Ref. [9], male-to-male dyads were observed to exchange brief handshakes while female-female dyads and male-female dyads exchanged greetings with relatively long contacts, composed of mutual-lip-kisses, mutual-face-contacts, embraces and hand-to-upper-body touches. While handshakes among women are now common especially in a business environment, the handshake experiment of this paper is limited to male-to-male handshakes, which follows the findings in [8, 9].

The contact areas and forces were obtained by force sensors, which the experimenter wore on different sections of his right hand. These force sensors are made from tactile pads that uses capacitive sensing principles to detect the change in capacitance upon the contact of the upper and lower tactile pads (FingerTPS, Pressure Profile Systems, USA). A male experimenter shook hands with 30 male subjects who are students and researchers at the National University of Singapore (NUS) and were paid for their participation. The demographics of the experimental subjects are shown on Table 1. The experimenter was trained to give a neutral handshake and wait for the handshake partner to initiate the contact. The subjects were instructed to give their normal handshakes. Fig. 1 shows the photograph of the locations and the naming convention where the force sensors are mounted. Approval for the handshake experimental protocol was granted by the NUS Institutional Review Board.

Table 1. Experimenter and Subjects' data

	Experimenter	Subjects (Mean ±Std Dev)
Age	22	26.35 ± 3.36
Height (cm)	175	173.42 ± 4.37
Weight (kg)	70	65.65 ± 8.57
Hand Length (mm)	190	187.50 ± 11.37
Hand Width (mm)	88	89.53 ± 6.80

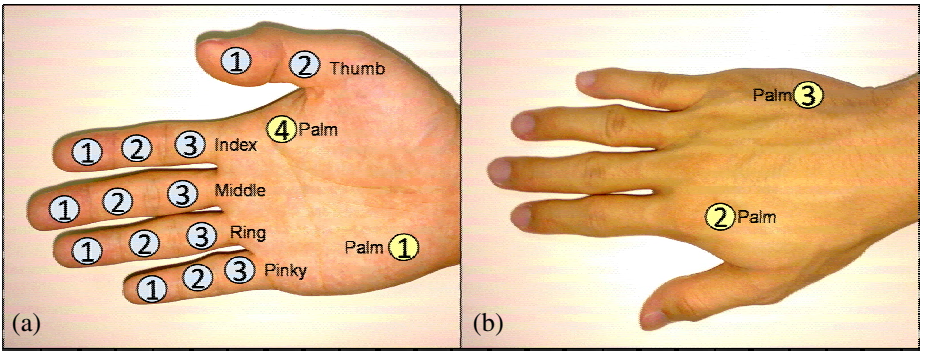


Fig. 1. (a) The areas where the sensors were placed on the hand of the male experimenter for the handshake experiment. The distal (i.e. fingertip), middle and proximal phalanges were given the number coding of 1, 2 and 3, respectively. Large area force sensors were also worn on dorsal part of the hand (i.e. opposite the palm; Fig 1b).

2.2 Results

The results are plotted on Fig. 2. Each sensorized area of the hand is shown in the x-axis while y-axis represents the maximum force read-outs from the tactile sensors. The areas where high contact forces are experienced (i.e. forces greater than 2 N) are at the palm, the thumb, the third phalange of the pinky finger and the middle phalanges of the pinky, ring and middle fingers. These phalanges are the locations where the full enclosure of the other person's hand can be achieved. For the purpose of indentation experiments, the middle phalanges of the pinky, ring and middle phalanges are selected, which are henceforth named Pinky2, Ring2 and Middle2.

3 Indentation Experiments

Indentation experiments are done on both the flat-hand position and on the curled-hand position. The curled-hand position, as shown in Fig. 3, represents the shape of the hand during handshake, while the flat-hand position represents the neutral position. The experiments are conducted on the selected phalanges where high contact forces occur during handshake. The objective is to obtain the force-displacement data which will serve as the benchmark data for duplication with synthetic skin materials, as well as to compare the force-displacement in the tested locations on curled-hand and on flat-hand positions.

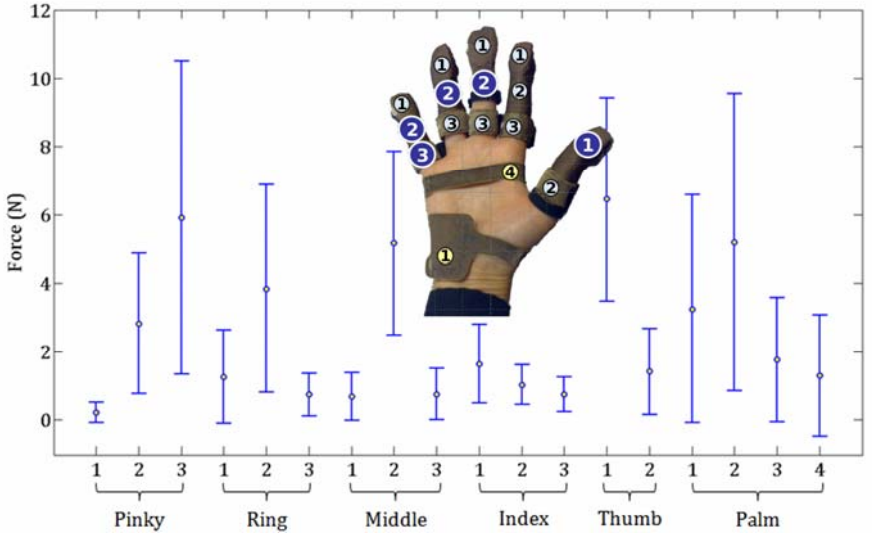


Fig. 2. Parts of the hand with the corresponding contact forces during handshake. The Inset shows the hand where the highlighted numbers show the phalanges which have contact forces greater than 2 N.

3.1 Methodology

A testing machine (MicroTester™, Instron, UK), attached with a load cell of limit 5 N, was employed to make indentations on different parts of the hand. A specially fabricated brass indenter (20 mm x 10 mm) was slotted into the load cell. For the flat-hand position, the hand was positioned above a rigid platform with the palm facing upwards. For the curled-hand position, the hand was placed on top of a specially made mould from gypsum so as to keep it at a normal handshake position. The

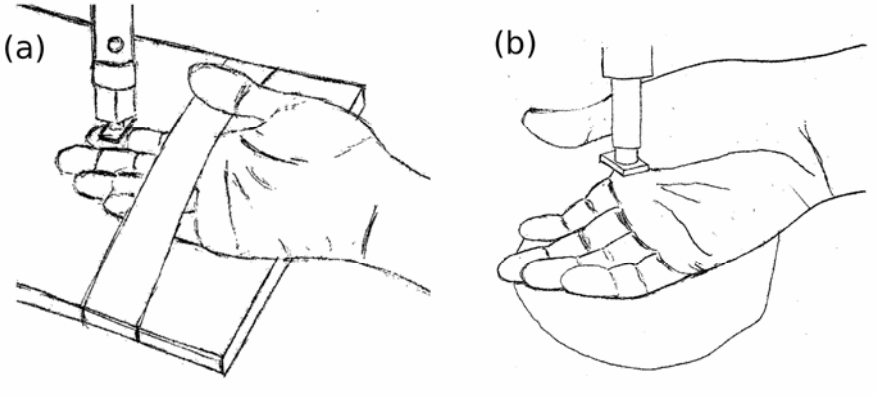


Fig. 3. Experimental set-up (a) flat position (b) curled position

platform was mounted on top of a stage that provides 25 mm translations on both the x and y axes for fine adjustments. The translational stage was in turn secured to the base of the testing machine. The set-up is shown on Fig. 3.

There are five test subjects each for both the flat-hand indentation and the curled-hand indentation. For each subject, all the test areas were indented with the indenter with a ramp speed of 0.5mm/s, with a compressive force of up to 4 N. The summary data on the experimental subjects are shown on Table 2. Approval for the indentation experimental protocol was similarly granted by the NUS Institutional Review Board.

Table 2. Subjects' data for the Indentation Experiment

	Subjects (Mean ±Std Dev)
Age	23.21 ± 0.84
Height (cm)	175.13 ± 5.79
Weight (kg)	68.14 ± 11.42
Hand Length (mm)	185.60 ± 10.06
Hand Width (mm)	85.22 ± 5.63

3.2 Results

The force–displacement curves obtained from the three test areas as well as the comparison between the displacement at 2 N force at the three test areas for both the flat-hand position and the curled-hand position are plotted on Fig. 4. Note that the 2 N force was set as the cut-off value that resulted into the highlighted phalange parts of the Fig. 2 Inset. The curves show that the skin tissues of the human finger phalanges

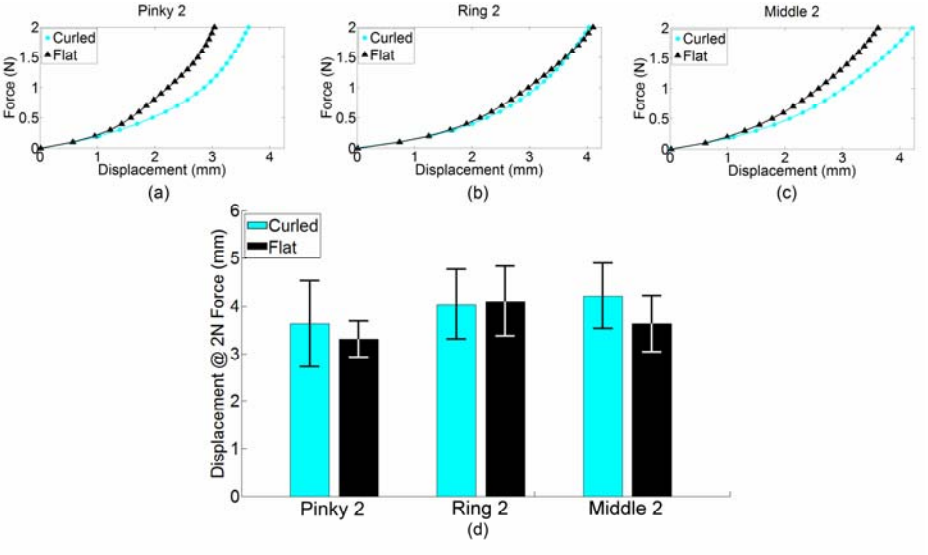


Fig. 4. Indentation test results at (a) Pinky2, (b) Ring2, and (c) Middle2, and (d) comparison of average displacement at the three phalanges at 2N applied force with flat and curled hand position. The error bars represent the standard deviation.

are compliant. The indentation results show that the compliance on second phalanges differs when the hand is in the flat position and in the curled position, a position that replicates the handshake grasp. The differences are significant in the Middle2 and Pinky2 area while the differences are slightly less obvious in the Ring2 area, as shown on Fig. 4.

4 Concluding Remarks and Future Improvements

In this paper, the areas of the hand where typical contact occurs during male-to-male handshakes were determined. The results show that the following areas of the hand have forces greater than 2 N when grasped during handshake: (a) the palm and the back of the palm, (b) the thumb, (c) the third phalange of the pinky finger and the (e) middle phalanges of the pinky, (f) ring and (g) middle fingers. These are the areas that envelope the handshaking partners' hands for a full grasp.

The middle phalanges of the pinky, ring and middle fingers were selected for indentations with a testing machine. The force-displacement curves were obtained on both flat hand position and curled-fingers position. The indentation results show that the skin tissues at these finger phalanges are compliant exhibiting high displacements with minimal forces applied. The results also show that there are differences in the force-displacement data on flat-hand position and curled-fingers position on the phalanges of the pinky, ring and middle fingers.

For future experiments, the quality of the mould used to support the hand in the curled position needs to be improved for the hand to be fixed better. The number of samples tested, which is currently only five for each indentation tests, will be increased as well so as to better represent the population.

Acknowledgments. This work is supported by the grant approved by the Faculty Research Committee from the Academic Research Fund Tier 1, Ministry of Education, Singapore. The authors are grateful to the technical contributions of Bharath Ramamoorthy and Bahareh Ghotbi.

References

1. Chang, S.O.: The Conceptual Structure of Physical Touch in Caring. Journal of Advanced Nursing 33, 820–827 (2001)
2. Crusco, A.H., Wetzel, C.G.: The Midas Touch: The Effects of Interpersonal Touch on Restaurant Tipping. Personality and Social Psychology Bulletin 10, 512–517 (1984)
3. Gueguen, N.: Kind of Touch, Gender and Compliance with a Request. Studia Psychologica 44, 167–171 (2002)
4. Cabibihan, J.J., Ge, S.S.: Towards Humanlike Social Touch for Prosthetics and Sociable Robotics: Three Dimensional Finite Element Simulations of Synthetic Finger Phalanges. In: International Conference on Social Robotics, Incheon, Korea (submitted, 2009)
5. Pillet, J., Didierjean-Pillet, A.: Aesthetic Hand Prosthesis: Gadget or Therapy? Presentation of a New Classification. The Journal of Hand Surgery 28, 523–528 (2001)

6. Cabibihan, J.J., Pattofatto, S., Jomaa, M., Benallal, A., Carrozza, M.C.: Towards Humanlike Social Touch for Sociable Robotics and Prosthetics: Comparisons on the Compliance, Conformance and Hysteresis of Synthetic and Human Fingertip Skins. International Journal of Social Robotics 1, 29–40 (2009)
7. Friedman, R.M., Hester, K.D., Green, B.G., LaMotte, R.H.: Magnitude Estimation of Softness. Exp. Brain Res. (2008)
8. Kendon, A., Ferber, D.: A Description of Some Human Greetings. In: Michael, R.P., Crook, J.H. (eds.) Comparative Ethology and Behavior of Primates, pp. 591–668. Academic Press, New York (1973)
9. Greenbaum, P.E., Rosenfeld, H.M.: Varieties of Touching in Greetings: Sequential Structure and Sex-Related Differences. Journal of Nonverbal Behavior 5, 13–25 (1980)

Towards Humanlike Social Touch for Prosthetics and Sociable Robotics: Three-Dimensional Finite Element Simulations of Synthetic Finger Phalanges

John-John Cabibihan and Shuzhi Sam Ge

Social Robotics Laboratory, Interactive and Digital Media Institute and
The Department of Electrical and Computer Engineering,
National University of Singapore, 4 Engineering Drive 3, Singapore 117576
{elecjj,elegesz}@nus.edu.sg

Abstract. Synthetic skins with humanlike characteristic would make it possible to address some of the psychosocial requirements of prosthetic hands as well as the safety and acceptance issues in social robotics. This paper describes the development of three-dimensional finite element models of synthetic finger phalanges. With the aim of duplicating the skin compliance of human finger phalanges, the model was used to investigate the effects of (i) introducing open pockets in the internal structure and (ii) combining different materials as external and internal layers. The results show that having pockets in the internal structure of the design can increase the skin compliance of the synthetic phalanges and make it comparable with the human counterpart. Moreover, having different layers can be used to satisfy skin compliance and other design requirements such as wear and tear.

Keywords: Social Robotics, Prosthetics, Artificial Skins, Rehabilitation Robotics, Finite Element Analysis.

1 Introduction

Social touches are those instances when we touch other people, such as shaking hands, hugging, patting on the shoulders. Endowing humanlike social touch for sociable robots and prosthetics would have positive benefits for both the human interaction partner and the prosthetics user. It was suggested that socially interactive robots must demonstrate 'believable' behavior: it must establish appropriate social expectations, it must regulate social interaction and it must follow social norms and conventions [1]. For prosthetic users, a prosthetic hand with humanlike appearance, motion, softness and warmth would allow the user's condition to be unnoticed by other people and shield the user from the social stigma and could lead to the speedy improvement of his or her emotional well-being (cf. [2-4]).

The duplication of humanlike softness for prosthetics and sociable robotics has been an ongoing theme of the authors' previous papers [5-8]. Through finite element (FE) simulations, the aim is to find a synthetic skin design that has similar characteristics with the human counterpart when indented. In accordance with this, the objective of this paper is to compare the softness properties of a synthetic finger

J.-H. Kim et al. (Eds.): FIRA 2009, LNCS 5744, pp. 80–86, 2009.

phalange design with the skin compliance data from handshake experiments. Softness has been previously defined as the perceptual correlate of skin compliance, which is the amount of deformation caused by an applied force [9]. In the accompanying paper [10], handshake experiments were conducted in order to obtain the locations where high contact forces occurred. Indentation experiments were also carried out to obtain the skin compliance data in these areas.

Unlike in the previous papers where a two-dimensional plane strain finite element model of a fingertip was utilized [5-7], this paper presents the development of a three-dimensional model. This model will be used to investigate the effects of (i) varying the internal topology of the finger phalange, (ii) varying different materials for the internal and external layers and (iii) comparing the skin compliance behaviors of the synthetic and human phalanges. The next section describes the modeling procedures: the materials and visco-hyperelastic constitutive equations are described, the geometry and the boundary conditions of the 3D FE finger phalange are presented and to validate the model, the simulation results of the 3D model were compared with the results of quasi-static indentation experiments. Then, the effects of using different internal topologies and using different material combinations are presented. Finally, the simulation results are compared with the compliance behavior of the human finger phalanges.

2 Finite Element Modeling and Simulations

2.1 Material Samples and Constitutive Equations

Samples of silicone (GLS 40, Prochima, s.n.c., Italy) and polyurethane (Poly 74-45, Polytek Devt Corp, USA) were previously characterized [5, 11]. The silicone sample has a Shore A value of 11 while the polyurethane sample has a value of 45 (i.e., a lower value indicates a low resistance to an indenter in a standard durometer test). These types of materials were selected as they were previously used as skins for prosthetics and robotics [12, 13]. The same visco-hyperelastic constitutive equations were used to represent the behavior of the synthetic materials as in the previous work [5]. The total stress is equal to the sum of the hyperelastic (HE) stress and the viscoelastic (VE) stress such that:

$$\sigma(t) = \sigma_{HE}(t) + \sigma_{VE}(t) \tag{1}$$

where t is the time. The hyperelastic behavior was derived from a function of strain energy density per unit volume, U.

$$U = \sum_{i=1}^{N} \frac{2\mu_i}{\alpha_i^2}\left[\lambda_1^{\alpha_i} + \lambda_2^{\alpha_i} + \lambda_3^{\alpha_i} - 3 + \frac{1}{\beta}(J^{-\alpha_i\beta} - 1)\right] \tag{2}$$

$$\sigma_{HE} = \frac{2}{J} F \frac{\partial U}{\partial C} F^{\mathrm{T}} \tag{3}$$

where $J = \lambda_1 \lambda_2 \lambda_3$ is the volume ratio, α_i and μ_i are the hyperelastic material parameters, $\beta = \nu/(1 - 2\nu)$ where ν is the Poisson's ratio, N is the number of terms

used in the strain energy function, and F and C are the deformation gradient and the right Cauchy-Green deformation tensors, respectively. It was assumed that the candidate materials were incompressible, and therefore J was set to unity. In the case of uniaxial compression tests, the following were used for the identification: $\lambda_1 = \lambda$, $\lambda_2 = \lambda_3 = \lambda^{(1/2)}$.

The viscoelastic behavior was defined as follows, with a relaxation function g(t) applied to the hyperelastic stress:

$$\sigma_{VE} = \int_0^t \dot{g}(\tau)\sigma_{HE}(t-\tau)d\tau \tag{4}$$

In order to describe several time constants for the relaxation, the stress relaxation function g(t) was defined using the Prony series of order N_G, where g_i and τ_i are the viscoelastic parameters:

$$g(t) = \left[1 - \sum_{i=1}^{N_G} g_i (1 - e^{-t/\tau_i})\right] \tag{5}$$

The number of coefficients to identify is equal to $2(N+N_G)+1(\nu)$ for each material type. The coefficients for hyperelastic (N), stress relaxation (N_G) and Poisson's numbers (ν) are given in Table 1.

2.2 Finite Element Model

Fig. 1 shows the three-dimensional geometry of the finger phalange that was modeled using the commercial finite element analysis software Abaqus™ / Standard 6.8-1 (Simulia, Providence, RI). The simulations were run at the Supercomputing and Visualisation Unit of the Computer Centre, National University of Singapore. Fig. 1 shows the geometry consisting of two layers. The external layer has a 0.8 mm thickness, which is comparable to the thickness of the epidermis/dermis skin layers of the human finger. The inner layer is designed to have different internal geometries. The Abaqus™ tetrahedral elements were used in conjunction with its automatic seed mesh feature. The base of the finger geometry was constrained in all degrees of freedom, which represents the bone structure of the human finger.

Table 1. Coefficients of the synthetic materials

i	1	2	3
Silicone ($\nu = 0.49$)			
g_i	0.015	0.044	0.029
τ_i (s)	0.025	0.150	0.300
μ_i (MPa)	0.080	0.010	-
α_i	0.001	15.500	-
Polyurethane ($\nu = 0.47$)			
g_i	0.167	0.158	0.113
τ_i (s)	0.100	1.380	25.472
μ_i (MPa)	0.100	0.063	-
α_i	5.500	8.250	-

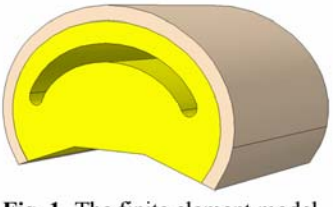


Fig. 1. The finite element model

2.3 Numerical Simulations

Simulations were conducted to determine the effects of varying the internal topology and material combinations in the skin compliance result of the synthetic finger phalanges. These will then be compared to the skin compliance of the human finger phalanges. The finger phalange width was 16 mm and the length was made to be 10 mm (i.e. towards the page). To investigate the effect of different internal structures, the internal layer was made to have three topologies: a solid internal geometry (Fig. 2a) and arc-shaped pockets with 1 mm (Fig. 2b) and 2 mm heights (Fig. 2c). To investigate the effects of different material combinations, the material coefficients of the external and internal layers can be changed. For example, to have a homogeneous solid material of silicone, the inner and outer layers have the same set of material coefficients; to have silicone as the inner layer and polyurethane as the outer layer, the material coefficients are changed accordingly. The 3D model was indented until a force of 1 N was reached. The loading rate was 0.5 mm/s. The simulation curves for the forces and displacements were plotted.

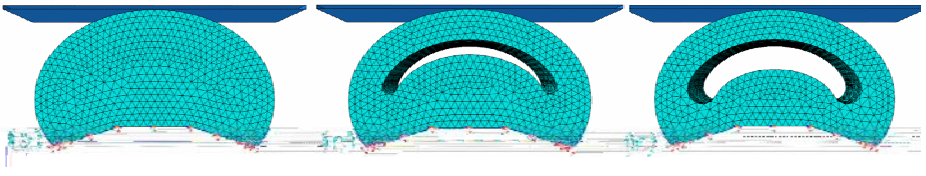


Fig. 2. Geometries and meshes of the 3D finite element models. (a) Solid internal geometry. Internal geometry pockets of 1 mm (c) and 2 mm (d).

3 Results and Discussion

Fig. 3 shows the results of the numerical simulations. Due to the limited space and for ease of comparisons, the simulation curves showing the effects of varying the internal geometry and material combinations are shown of Fig. 3a. The results are summarized in Table 2.

Fig. 3b shows the design configuration that is closest to the skin compliance of the human finger phalanges. The simulation results are from the models with internal pockets of 2 mm height. The results are summarized in Table 3.

Effect of Pockets in the Internal Structure. Shown on the leftmost cluster of curves in Fig. 3a are the simulation results of the models with solid internal geometries. A 1 N compressive load resulted into 0.23 mm displacement for polyurethane (PU) and a 0.46 mm displacement for silicone (SIL). Introducing a 1 mm height pocket increased the displacement to 1.18 mm for PU and 1.42 mm for SIL. These correspond to 413% and 209% increase in the displacement values, respectively. Having a 2 mm height pocket gives displacements of 2.05 mm for PU and 2.29 mm for SIL, corresponding to 791% and 398% increase, respectively, from the solid internal geometry configuration. These results show that having internal pockets can significantly increase the skin compliance results of synthetic skins.

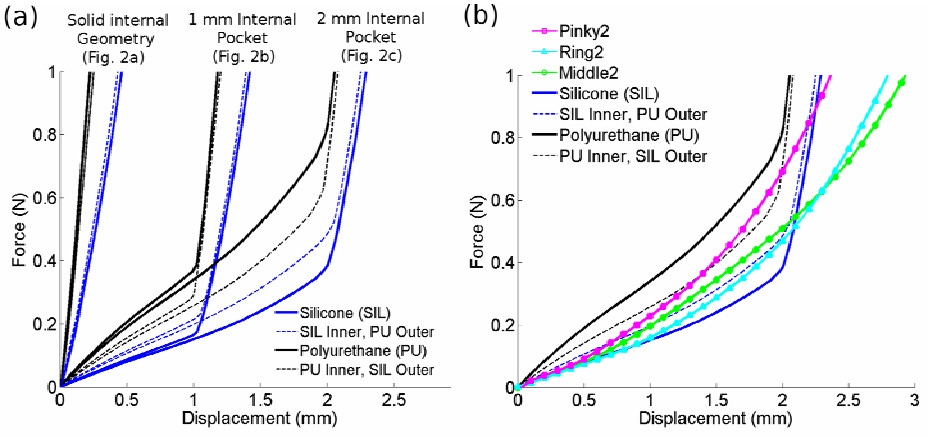


Fig. 3. Results of the numerical simulations. (a) Simulation results of varying the topology of the internal layer and material combinations. (b) Comparisons between the results of the 2 mm inner pocket and human finger phalanges.

Table 2. Displacement (mm) at 1 N for Internal Topology and Material Combinations

	Solid Internal	1 mm Pocket	2 mm Pocket
Polyurethane (PU)	0.23	1.18	2.05
Silicone (SIL)	0.46	1.42	2.29
PU inner – SIL outer	0.25	1.20	2.07
SIL inner – PU outer	0.44	1.39	2.25

Effect of Changes in Material Combination. Shown as the thin, dashed lines and clustered at the leftmost part of Fig. 3a are the simulation results of the solid internal geometry configuration. A 1 N compressive load for a combination of PU inner layer and a 0.8 mm thin outer layer of SIL resulted into a displacement of 0.25 mm, or an 8.7% increase from a homogeneous PU material condition. With a combined inner layer of SIL and outer layer of PU, the displacement was 0.44 mm or a 4.3% decrease from a SIL-only material condition. These results are understandable because the SIL material has a lower durometer value (i.e. softer) as compared to the PU material (cf. Sec 2.1).

For the remaining combinations, the changes in the displacements at the 1 N compressive load correspond to an increase or decrease of displacement values to within 20% difference. It can also be seen from Fig. 3a that there are significant changes in the slope of the loading curves due to variations in the material layers, particularly for the geometries with the 2 mm pocket configurations. For the 'PU inner and SIL outer' condition, notice that the slope of the loading curve is lower as compared to the 'PU-only' homogeneous material configuration. At the 2 mm displacement, the force is reduced from 0.75 N for the 'PU-only' configuration to the 0.63 N result for the 'PU inner and SIL outer' configuration (i.e. about 16% decrease).

For the 'SIL inner and PU outer' configuration, the force values at the 2 mm displacement are 0.39 N for the 'SIL-only' configuration and 0.5 N for the 'SIL inner and PU outer' configuration or a 28% increase in force value. Having a 0.8 mm outer layer affects the slope of the loading curve resulting into an increase or decrease of the compressive forces. These occur before the top layer of the pocket comes in contact with the bottom layer of the pocket and eventually stiffens.

Comparisons with Human Finger Phalanges. Fig. 3b shows the skin compliance behavior of the middle phalanges of the pinky, ring and middle fingers. These are the parts of the human hand which experiments have shown to have high contact forces during handshake interactions [10]. It can be observed that the synthetic finger phalange configurations with 2 mm inner pockets can approximate the human skin compliance, particularly the middle phalange of the pinky finger (i.e. Pinky2). Table 3 shows that at 1 N compression load, SIL has a displacement of 2.29 mm as compared to 2.15 mm of the Pinky2 finger phalange. The 'PU inner – SIL outer' configuration is closest to the Pinky2 curve until 0.6 N. The 'SIL inner – PU outer' curve approximates Ring2 and Middle2 curves until 0.5 N. Below 0.4 N, the SIL only curve is more compliant than the three human finger phalanges that were studied. The curves of the human phalanges diverge from the results of the synthetic finger phalanges beyond the 1 N load.

Table 3. Skin Compliance Results

	Displacement (mm) at 1 N
Pinky2	2.15
Middle2	2.80
Ring2	2.94
Polyurethane (PU)	2.05
Silicone (SIL)	2.29
PU inner – SIL outer	2.07
SIL inner – PU outer	2.25

4 Concluding Remarks

This paper presented a 3D finite element model for investigating the effects of varying the internal topology and the composition of material layers in an attempt to duplicate the skin compliance of human finger phalanges. The following conclusions can be made. First, the skin compliance can be increased by introducing open pockets on the internal structure of a synthetic finger phalange. Second, having one type of material for the 0.8 mm external layer and another type for the internal layer can affect the deflection of the finger phalanges' surface, but this combination has minimal effect when the top layer of the pocket comes into contact with the base of the finger phalange. This finding can be used to satisfy requirements for skin compliance and other requirements such as wear and tear. For instance, having a stiff external layer and a softer internal layer could satisfy the practical requirements. Lastly, the simulation results show that the synthetic skins with the configurations described herein could achieve humanlike skin compliance.

Acknowledgments. This work is supported by the grant approved by the Faculty Research Committee from the Academic Research Fund Tier 1, Ministry of Education, Singapore.

References

1. Fong, T., Nourbakhsh, I., Dautenhahn, K.: A Survey of Socially Interactive Robots. Robotics and Autonomous Systems 42, 143–166 (2003)
2. Murray, C.D.: The Social Meanings of Prosthesis Use. Journal of Health Psychology 10, 425–441 (2005)
3. Leow, M.E.L., Pho, R.W.H., Pereira, B.P.: Esthetic Prostheses in Minor and Major Upper Limb Amputations. Hand Clinics 17, 489–497 (2001)
4. Pillet, J., Didierjean-Pillet, A.: Aesthetic Hand Prosthesis: Gadget or Therapy? Presentation of a New Classification. The Journal of Hand Surgery 28, 523–528 (2001)
5. Cabibihan, J.J., Pattofatto, S., Jomaa, M., Benallal, A., Carrozza, M.C.: Towards Humanlike Social Touch for Sociable Robotics and Prosthetics: Comparisons on the Compliance, Conformance and Hysteresis of Synthetic and Human Fingertip Skins. International Journal of Social Robotics 1, 29–40 (2009)
6. Cabibihan, J.J.: Design of Prosthetic Skins with Humanlike Softness. In: Lim, C.T., Goh, J. (eds.) Intl. Conference on Biomedical Engineering, vol. 23. Springer, Singapore (2008)
7. Cabibihan, J.J., Ge, S.S.: The Influence of Internal Geometry Modifications on the Skin Compliance Behavior of an Anthropomorphic Fingertip. In: Intl. Conference on Humanoids, Nanotechnology, Information Technology, Communication and Control, Environment and Management, Manila (2009)
8. Cabibihan, J.J., Carrozza, M.C., Dario, P., Pattofatto, S., Jomaa, M., Benallal, A.: The Uncanny Valley and the Search for Humanlike Skin Materials for a Prosthetic Fingertip. In: Proceedings of IEEE-RAS Humanoid Robotics, Genova, Italy, vol. 1, pp. 474–477 (2006)
9. Friedman, R.M., Hester, K.D., Green, B.G., LaMotte, R.H.: Magnitude Estimation of Softness. Exp. Brain Res. (2008)
10. Cabibihan, J.J., Pradipta, R., Chew, Y.Z., Ge, S.S.: Towards Humanlike Social Touch for Prosthetics and Sociable Robotics: Handshake Experiments and Finger Phalange Indentations. In: International Conference on Social Robotics, Incheon, Korea (submitted, 2009)
11. Cabibihan, J.J., Pattofatto, S., Jomaa, M., Benallal, A., Carrozza, M.C., Dario, P.: The Conformance Test for Robotic/Prosthetic Fingertip Skins. In: Proceedings of the First IEEE/RAS-EMBS International Conference on Biomedical Robotics and Biomechatronics, vol. 1, pp. 561–566 (2006)
12. Edin, B.B., Ascari, L., Beccai, L., Roccella, S., Cabibihan, J.J., Carrozza, M.C.: Bio-Inspired Sensorization of a Biomechatronic Robot Hand for the Grasp-and-Lift Task. Brain Research Bulletin 75, 785–795 (2008)
13. Beccai, L., Roccella, S., Ascari, L., Valdastri, P., Sieber, A., Carrozza, M.C., Dario, P.: Development and Experimental Analysis of a Soft Compliant Tactile Microsensor for Anthropomorphic Artificial Hand. IEEE/ASME Transactions on Mechatronics 13, 158–168 (2008)

Investigating the Effects of Gain and Loss of Esteem on Human-Robot Interaction

Takanori Komatsu[1], Hideaki Kaneko[2], and Takashi Komeda[2]

[1] Shinshu University
Tokida 3-15-1, Ueda, Nagano 386-8567, Japan
tkomat@shinshu-u.ac.jp
[2] Shibaura Insitute of Technology
Fukasaku 307, Mimuna, Saitama 370-8567, Japan
{m407019@,komeda@se}.shibaura-it.ac.jp

Abstract. The purpose of this study was to investigate the relationship between the robot's static and dynamic design experimentally, and to compare the results of this experiment with the findings of the gain and loss of esteem. Here, the static design is about the appearance or configuration of the robot, and the dynamic design is about the trajectories, behaviors of the robot and achievement level of the given tasks. The results showed that the static design did not have effects on the participants' impressions, but the dynamic design did have strong effects on their impressions.

Keywords: gain and loss of esteem, static/dynamic design, verbal instruction.

1 Introduction

Recently, various kinds of home-care robots have been developed to assist our daily lives, and some of them are actually commercialized [1]-[3]. Therefore, these robots would be much closer to people in public space. The traditional robotics technologies have been sophisticated especially in production sites, and these developed industrial robots are quite good at assembling the machinery products or cutting certain parts promptly and precisely. On the other hands, the robotics technologies required for home-care robots are completely different with the ones for the industrial robots. These robots are not really required to behave promptly and precisely but to have familial design of their appearance because these robots should not make users feel dangerous and anxious.

In order to consider the robot design, we assumed that there are at least two design perspectives [4]; one is static design which is about the appearance or configuration of the robot that is the same meaning with the general terms "design," and the other is dynamic design which is about the trajectories, behaviors of the robot and achievement level of the tasks. We assumed that the robots' users would determine their impressions of the robot based on these two design perspectives. Currently, humanoid type robots are well applied to home-care usage because these robots are believed to have the effective static and dynamic designs. We also agreed that such robots could provide very familial and sophisticate impressions to users firstly.

J.-H. Kim et al. (Eds.): FIRA 2009, LNCS 5744, pp. 87–94, 2009.

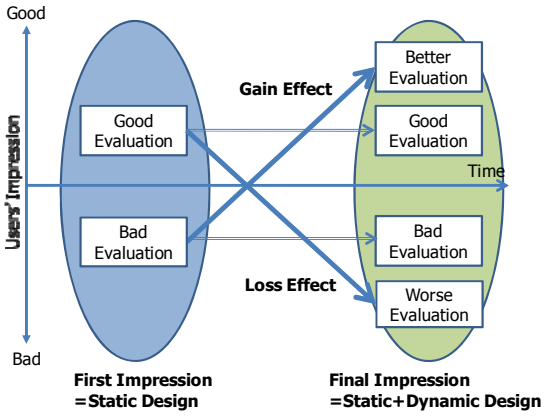


Fig. 1. The relationship between "gain and loss of esteem" and "static/dynamic design"

However, there are some possibilities that serious problems would occur when these robots failed to achieve the given task. For example, when a user is living together with a humanoid robot and this robot behaves as s/he intended, s/he would satisfy this robot and would think that this is usual situation because the robot's appearance is familiar to users. However, this robot does not behave as s/he intended, s/he would be surprised and eventually disappointed with this robot deeply. This accident might affect the users' impressions of the robot; that is, the impressions would be getting worse.

This example could be explained by the gain and loss of esteem [5] which is the one of the significant findings in social psychology; that is, *person A prefers person B, who changed her/his evaluation of person A from positive to negative to person C, who maintained a positive evaluation (gain effect), and person A has a worse perception of person B, who changed her/his positive evaluation into a negative one than person C, who maintained a negative evaluation (loss effect; the above example would be this case)*. Here, the first impressions of others in gain and loss of esteem correspond with the static design in human-robot interaction, while the final impressions of others do with the static and dynamic design (Fig. 1). Therefore, if the relationship between the static design and dynamic design was the similar to the one between the first and final impression in gain and loss of esteem, the importance of considering the static and dynamic design for developing the home-care robot would be clarified.

The purpose of this study is then to investigate the relationship between the static and dynamic design experimentally, and to compare the results of this experiment with the findings of the gain and loss of esteem. The results of this study would contribute to proposing the effective design methodology of home-care robots required living together with users.

2 Experiment

2.1 Settings

21 Japanese undergrads (10 men and 11 women; 18 – 24 years old) participated in this experiment, and they were randomly assigned to the following four experimental conditions.

- **Condition 1 (6 participants)**: Good static design with good dynamic design
- **Condition 2 (5 participants)**: Good static design with bad dynamic design
- **Condition 3 (5 participants)**: Bad static design with good dynamic design
- **Condition 4 (5 participants)**: Bad static design with bad dynamic design

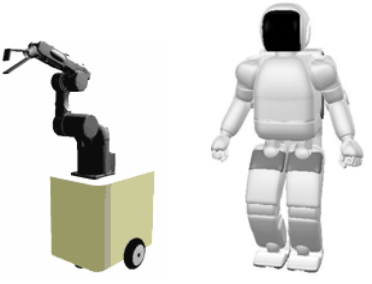


Fig. 2. AMOS robot (left), and a humanoid type robot (right)

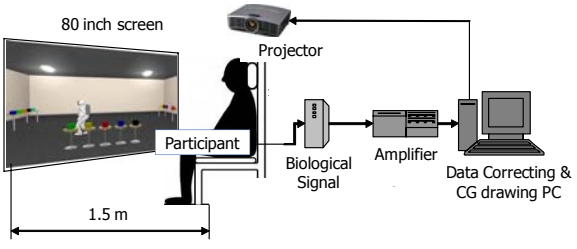


Fig. 3. Experimental setting

As a robot with "good static design (Condition 1 and 2)" we prepared a humanoid robot, while as a robot with "bad static design (Condition 3 and 4)" we prepared a human-care robot "AMOS (Assistive Mobile Robot System)" [6] (Fig. 2). And as a "good dynamic design" we made the robot succeed in behaving according to the user's order, while as a "bad dynamic design" we did the robot fail to behave as the user's order. Actually, these robots were prepared as computer graphics, and these robots moved around in the virtual world on 80-inch screen (Fig. 3). Then, if the relationship between the static and dynamic design was similar to the findings of the gain and loss of esteem, it is expected that the Condition 3 showed the higher evaluation than Condition 1, and the Condition 2 showed the lower evaluation than Condition 4.

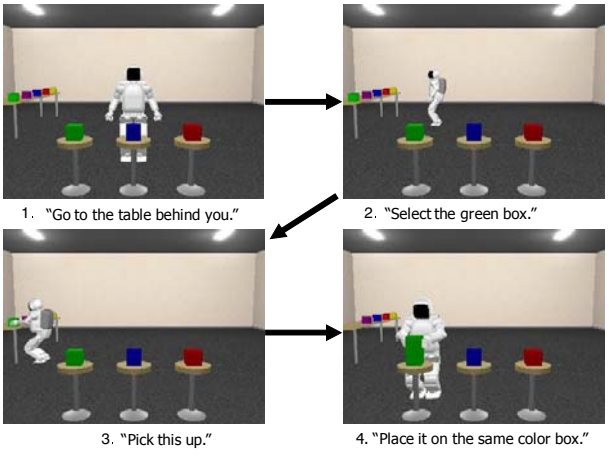


Fig. 4. The procedure of the verbal instructions asked the participants to give

2.2 Procedure

Firstly, the experimenter told the participants as "The projected robot is a home-care robot which is developed in our laboratory. Your task is to give the verbal instruction to this robot, and to evaluate this robot." The participants were asked to give the verbal instruction to the robot in order to bring the color box which the participants intended. The participants experienced this task three times. One task took about 80 seconds, so that the total time of this experiment took about 5 minutes. The detailed instruction step of this task is depicted in Fig. 4.

In Condition 1 and 3 (both with good dynamic design), the robots behaved as the participants intended; that is, these robot succeeded in bring the color box which the participants ordered. On the other hand, in Condition 2 and 4 (with bad dynamic design), the robot failed to behave as the participants intended, e.g., taking the wrong box or drop it.

2.3 Acquired Data

In order to comprehend the participants' impressions of the robot, we investigated the participants' subjective impressions of the robot by means of questionnaire. As a questionnaire to comprehend the participants' subjective impressions, we prepared a questionnaire with 20 questions. Each question is 10 point likert scale (maximum score is 10 point, and minimum score is 1 point). The participants were asked to fill in this questionnaire before and after the experiment (Fig. 5). And we also measured the several biological signals as objective indicators; that is, heart beat variability (LF/HF ratio), electrodermal activity, and amylase monitoring stress check. The higher LF/HF ratio as heart beat variability, the higher voltage of EDA, and the higher values of amylase monitor indicated that the participants feel stronger mental stress.

Q1	I think that this robot is very friendly
Q2	I want to have this robot.
Q3	I have very positive impressions on this robot
Q4	I have confidential impressions on this robot
Q5	I think that this robot will succeed in achieving this
Q6	I could commit the task to this robot
Q7	I think that this robot is very clever
Q8	I could easily ask to take this task
Q9	I think that this robot could obey my order
Q10	I think that this robot has own wills
Q11	I think that the elder person can use this robot
Q12	I want to be with this robot
Q13	I think that this robot have a higher adaptability
Q14	I think that this robot could take serious tasks
Q15	I think that this robot is quite good one
Q16	I have a lot of confidence with this robot
Q17	I think that all persons like this robot
Q18	I could recommend this robot
Q19	I think that everybody could use this robot
Q20	I think that this robot is quite sophisticated

Fig. 5. Questionnaire utilized in this experiment

3 Results

Questionnaire: Fig. 6 showed the results of questionnaire acquired just before the experiment, while Fig. 7 did the ones acquired just after the experiment. From Fig. 6, we could confirm that the significant differences were observed in Q7 and Q15 before the experiment. Therefore, it can be said that the static design did not have strong effects on the participants' impressions about the robots. And from Fig. 7, we could confirm that the significant differences were observed in Q1, 2, 3, 4, 5, 6, 8, 9, 10, 11, 12, 14, 16, 17, 18, 19 and 20 after the experiment. In these 17 questions, the values of the condition 1 and 3 (good dynamic design) showed the higher than one in condition 2 and 4 (bad dynamic design). However, there were significant different differences between the condition 2 and 4 in only Q17 and Q20. Therefore, the results of the questionnaire succeeded in showing the significant effects of dynamic design, but did not clearly show the gain and loss of esteem.

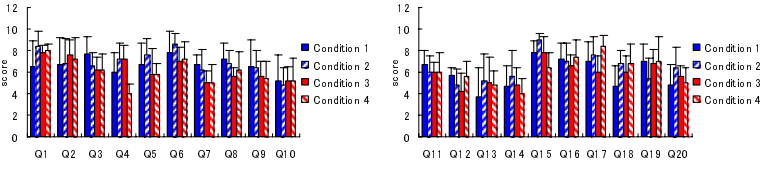


Fig. 6. The results of questionnaire before the experiment

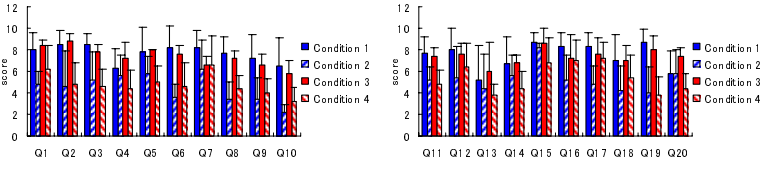


Fig. 7. The results of questionnaire after the experiment

Heart beat variability: Fig. 8 showed the LF/HF ratio acquired before (left) and during (right) the experiment. From these figures, we could not find any significant differences between these four conditions in before and during the experiment.

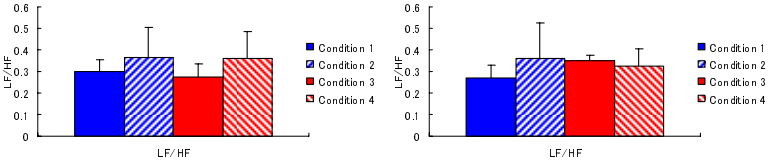


Fig. 8. Acquired the heart beat variability (LF/HF) before (left) and during (right) the experiment

Electrodermal Activity (EDA): Fig. 9 showed the EDA value acquired before (left) and during (right) the experiment. From these figures, we could find the significant tendencies between the (condition 1, 2) and (condition 3, 4) in before the experiment ($F(1,17)=1.17$, $p<.1$ (+)). It is then said that the participants who looked at the bad static design robot before the experiment had some mental stresses but this tendency were not shown during the experiment.

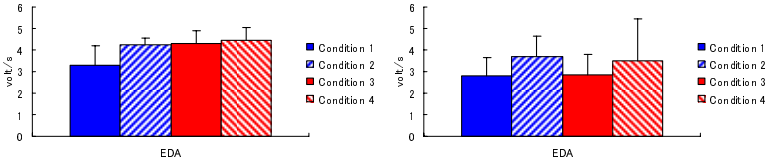


Fig. 9. Acquired the EDA value before (left) and during (right) the experiment

Amylase Monitor: Fig. 10 showed the value of amylase monitor acquired before (left) and after (right) the experiment. From these figures, we could find the significant differences between the Condition 2 and the other conditions after the experiment ($F(1,17)=3.82$, $p<.05$ (*)). Here, it is said that the participants who looked at the good static design robot which showed the bad dynamic design felt stronger mental stresses than the other conditions.

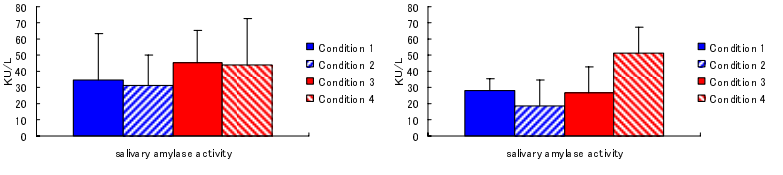


Fig. 10. Acquired the values of amylase monitor before (left) and after (right) the experiment

4 Discussion and Conclusions

The results of the experiment could be summarized as follows.

- The results of heart beat variability did not show any significant differences between the four experimental conditions in before and during the experiment. On the other hand, the results of EDA showed that there was a significant tendency between (condition 1, 2) and (condition 3, 4), and the ones of amylase monitor showed that there was a significant differences between condition 2 and the other conditions. Therefore, these objective indicators could not show that the gain and loss of esteem was observed in this experiment.
- The results of questionnaire showed that there were the significant differences between the experimental conditions before the experiment only in two out of 20 questions, while there were 17 out of 20 questions after the experiment. Here, it is then said that these 17 questions showed that values of the condition 1 and 3 (good dynamic design) showed the higher than one in condition 2 and 4 (bad dynamic design). However, there were significant different differences between the condition 2 and 4 in only Q17 and Q20. Therefore, the results of the questionnaire succeeded in showing the significant effects of dynamic design in the participants' impressions of the robot, but did not clearly show that the gain and loss of esteem was observed.

Therefore, it can be said that the results of the experiment showed that the gain and loss of esteems were not really observed. However, the importance of the dynamic design was revealed from the results of amylase monitor and the ones of the questionnaires. This means that the bad dynamic design regardless of the types of the static design induced the participants' lower impression, and good dynamic design did the higher impressions. On the other hand, it seemed that the types of the static design did not have effects on the participants' impressions about the robot. Therefore, the results of this experiment could be depicted like Fig. 11.

To compare this Fig. 11 with the Fig. 1, there is a significant difference especially in the participants' impressions before the interaction, but not really in the final impressions which is affected by the dynamic design. This comparison clearly

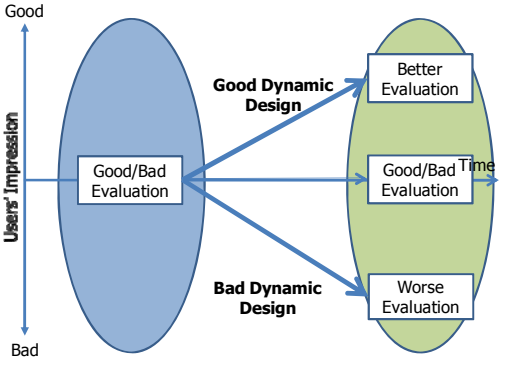


Fig. 11. Conceptual diagram of the results of this experiment

indicated that the impressions of the static design do not have much effect on the participants' impressions of the robot, but the dynamic design does have much effect.

About the less effects of the static design, Komatsu and Yamada [10] already argued the first impressions of the robots including the expected functions of the agents are generally diverged among the people since the impressions of the robots are determined by each people's mental model, and this mental model is constructed by each people's preferences and experiences about the robots. Therefore, the reason that the static design of AMOS and a humanoid robot did not show any differences would be caused by these diverged participants' first impressions. We expected that this result would have an impact of the design methodology of the home-care robot, like "Do focus on the dynamic design but on the static design."

As the follow-up study, we are also planning to tackle with the issue "how the impressions of the robot' static design are determined." To tackle with this issue, it is expected to include the findings of the social psychologies, like "impression formation [12]" or "interpersonal cognitions [13]." Although the investigation this issue would be quite tough works, we thought that it is worthwhile to conduct this follow-up study.

References

1. Kawamoto, H., Kanbe, S., Sankai, Y.: Power Assist Method for HAL-3 Estimating Operator's Intension Based on Motion Information. In: 12th IEEE Workshop on Robot and Human Interactive Communication, ROMAN 2003 (2003)
2. Shibata, T., Mitsui, T., Wada, K., Tanie, K.: Subjective Evaluation of Seal Robot: Paro - Tabulation and Analysis of Questionnaire Results. Journal of Robotics and Mechatronics 14(1), 13–19 (2002)
3. Davis, M., Dautenhahn, K., Nehaniv, C.L., Powell, S.D.: The narrative construction of our (social) world: steps towards an interactive learning environment for children with autism. Universal Access in the Information Society 6(2), 145–157 (2006)
4. Ohtsuka, S., Komatsu, T., Komeda, T.: Robot Design with Affinity – Evaluation of Functional Design and the Interaction by the Voice Instruction that Used Virtual Robot. Transactions of Human Interface Society (In Japanese) 8(3), 23–33 (2006)
5. Aronson, R., Linder, D.: Gain and loss of esteem as determinants of interpersonal attractiveness. Journal of Experimental Social Psychology 1, 156–171 (1965)
6. Takahashi, Y., Komeda, T., Koyama, H.: Development of the assistive mobile robot system AMOS-to aid in the daily life of the physically handicapped. Advanced Robotics 18(5), 473–496 (2004)
7. Taguchi, T.: Evaluation of Fatigue during Car Driving. The R&D Review of Toyota CRDL (In Japanese) 33(4), 25–31 (1998)
8. Ebe, K., Okuwa, M., Inagaki, H.: Evaluation of Driver's Mental Workload Due to Visual and Auditory Cognition. The R&D Review of Toyota CRDL (In Japanese) 34(4), 55–62 (1999)
9. Atsumi, B.: Evaluation of Mental Condition on Drivers by Analysis of Heart Rate Variability. In: 1994 JSAE Annual Congresses (In Japanese), pp. 133–136 (1994)
10. Komatsu, T., Yamada, S.: Effects of Adaptation Gap on Users' Differences in Impressions of Artificial Agents (submitted)
11. Anderson, N.H.: Application of an additive model to impression formation. Science 138, 817–818 (1962)
12. Baldwin, M.W. (ed.): Interpersonal Cognition. Guilford Press, New York (2006)

A Simple Momentum Controller for Humanoid Push Recovery

Bassam Jalgha and Daniel C. Asmar

Mechanical Engineering Department,
American University of Beirut, Beirut, Lebanon
da20@aub.edu.lb

Abstract. While working in a dynamic environment, humanoid robots are subject to unknown forces and disturbances, putting them at risk of falling down and damaging themselves. One mechanism by which humans avoid falling under similar conditions is the human momentum reflex. Although such systems have been devised, the processing requirements are too high to be implemented on small humanoids having microcontroller processing capabilities. This paper presents a simplified momentum controller for fall avoidence. The system is tested on a simulated robot developed under Gazebo as well as under a real humanoid. Results show successful fall avoidance.

1 Introduction

Bipedalism is unstable by nature, it is an active process that requires constant adjustment of balance. Millions of years of evolution have allowed humans to master and improve their balance strategies. Without any conscious decision making, we are able to easily maintain stability and withstand virtually any unexpected disturbance using our reflexes.

Research in the stability of humanoid robots is focused on two different tracks including dynamic gait stability (humanoid in motion), and stance stability (humanoid standing in place). Stance stability problems typically involve recovering from a disturbance while standing in place. Human beings deal with such disturbances via their reflexes, and take corrective actions such as lunge steps, rotation of the arms, or leaning on fixed objects.

Pratt *et al.* [7] [8] investigated the problem of push recovery of a humanoid in a standing stance, resting on one foot with the other one raised. They introduce what are known as Capture Points (CP), which are the locations on the ground where the robot can step into in order to avoid falling by taking only one step. In their work, the calculation of a CP is initially inferred from an inverted pendulum model of a robot [7] and then is refined via a learning algorithm [8]. Abdallah *et al.* [3] also work on the problem of push recovery, albeit with both feet grounded. In their work, they develop a methodology for fall avoidance by mimicking one of the human's reactive reflexes against falling.

J.-H. Kim et al. (Eds.): FIRA 2009, LNCS 5744, pp. 95–102, 2009.

When one is pushed above the torso with a moderate force, he/she attempts to mediate this disturbance by rotating the upper body about the hips and bending the lower body back as shown in Fig. 1. The physical interpretation of this reflex amounts to an increase in the angular momentum of the upper body as well as a reactive linear momentum such that the center of gravity of the body remains within the support (two feet) of the person [4]. The controller that is devised uses angular and linear momentum as criteria to stabilize the robot. Like all aspects of bipedal robots, online computations are difficult and complex due to the non-linearity and high dimensionality of the system. In the model of Abdallah *et al.* [3] fall avoidance includes a disturbance absorption strategy paired with a posture recovery scheme. The humanoid model they use includes three degrees of freedom and uses an elaborate scheme for calculating momenta.

The complexities of the aforementioned systems necessarily require computations that surpass the capabilities of a simple microcontroller. Indeed, the systems are implemented either on a simulator hosted on a Personal Computer or on an embedded PC housed on board the humanoid. This paper investigates simplifying the fall avoidance scheme by putting forward an simplified model, which is inspired by one of human's fall avoidance reflexes. Looking more closely at Fig. 1 one can notice that when humans are pushed with a moderate force, their first reactive action can include locking the knees and pivoting only about the hips and ankles. By mimicking this behavior, one is able to reduce the dimensionality of the problem from three degrees of freedom to two. By further assuming the humanoid's mass as concentrated in its upper body a relatively simple control strategy can be devised for push recovery. Towards this end, the focus of this paper is to investigate this hypothesis and to test it on a simulated humanoid robot developed under Gazebo [6] (Player/Stage/Gazebo project [1]), as well as on a real humanoid robot.

The remainder of this paper is structured as follows. Section 2 quickly reviews the momentum reflex and formulates the distrubance absroption model. Section 3 presents the feedback controller developed for disturbance absorption and compares it to the one introduces by Abdallah *et al.* [3]. Section 4 describes the experiments that validate the proposed system and discusses the results. Section 5 concludes and describes the direction future work.

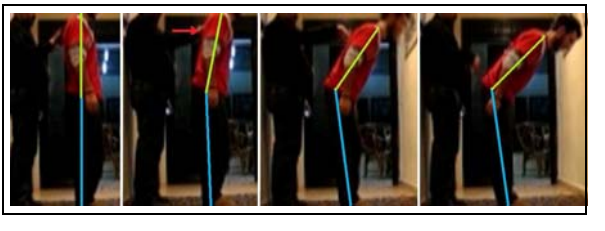


Fig. 1. Video images showing the human momentum reflex of a person undergoing a push disturbance

2 Understanding the Momentum Reflex

Sardain and Bessonnet [9] introduce the Center of Pressure (CoP) as the point where the resultant reaction of the ground is normal and the Zero Moment Point (ZMP) as the location where the summation of moments about it is equal to zero. The theory explains that for stability, the ZMP must always be kept inside the convex hull of the foot support polygon. Under these conditions the ZMP coincides with the CoP [9],[10]. According to Vukobratovic [10], as soon as the ZMP exceeds the foot limit, it would be called a fictitious ZMP since it would no longer be associated with the CoP, and a tipping moment would be initiated. Once the CoP reaches a boundary, it can no longer compensate for a disturbance by shifting its weight in that given direction and therefore looses one degree of actuation [3][7]. Abdallah *et al*. [3] argue that it is not stability that one should be concerned with but rather controllability, where they state that a robot could maitain a flat foot and still hit the ground with its trunk. Controllability is the ability of a controller to reach a final state from any given initial state. In their work, when a humanoid is pushed it avoids falling and recovers to its initial posture by taking two independent actions; first it follows a disturbance absorption scheme, second it follows a postures recovery scheme. For disturbance absorption a controller is devised, which uses the linear and angular momenta of the robot as input variables to control the stability of the robot. In their model, the robot is assumed to have three degrees of freedom, capable of pivoting about its ankles, knees, and hips. Our method diverges at this point from those discussed above based on the following argument.

If one closely observes how humans react to a push disturbance, one notices a locking of the knees while the upper and lower body-parts pivot. The implication of this behavior on a humanoid robot is a reduction of the degrees of freedom from three to two, with the knee joint remaining stiff. If we can further assume that the robot's mass is concentrated above its waist, the robot can be modeled as shown in Fig. 2.

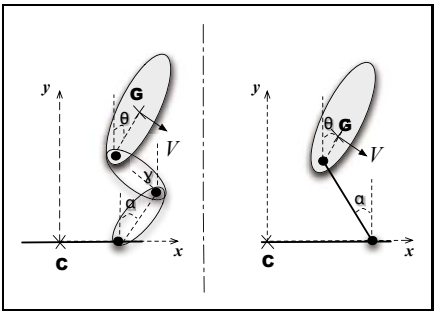


Fig. 2. Typical humanoid model (left) with 3DOF and links possessing mass, whereas the proposed model (right) has 2DOF and mass concentrated at the top link

In what follows, the mathematical model is briefly introduced followed by an explanation of the momentum controller. Posture recovery is not addressed in this paper because it is not within the scope of the contribution. The robot will simply be reset to its starting stance each time the disturbance absorption scheme is tested.

2.1 Mathematical Model

Applying Newton's second law for rotational motion about a random point C located on the ground near the base of the humanoid (See Fig. 3) one obtains:

$$\sum T_C = \sum T_{eC} + \boldsymbol{C_i} + \boldsymbol{r_{CG}} \times F_i = 0, \tag{1}$$

which by D'alembert's principle can be stated as the summation of torques T_C about a point C is equal to the summation of external torques T_{eC} plus the inertial couple C_i plus the torque caused by the inertial force F_i. If we restrict our analysis to a planar robot in 2D and rearranging as proposed in [3] we get:

$$\begin{aligned} x_p &= x_G + \frac{\dot{H}_G - y_G \dot{L}_x}{d} + \frac{y_F F_{dist}}{d} \\ d &= \dot{L}_y + mg, \end{aligned} \tag{2}$$

where x_P, and x_G are the horizontal distances from C to the center pressure P and center of mass G respectively, $\dot{H}_G$ is the change in angular momentum about point G, y_F is the vertical distance from C to the point of application of the applied disturbance F_{dist}, g is the gravitational acceleration. $\dot{L}_x$ and $\dot{L}_y$ are the x and y components of the time derivative of the linear momentum.

It is apparent from (2) that by regulating the change in angular momentum $\dot{H}_G$ and linear momentum $\dot{L}_x$ one can maintain the position of the CoP within the support region (*i.e.*, the convex hull formed by the two feet of the robot) and subsequently avoid having the robot fall. For steady static conditions, the

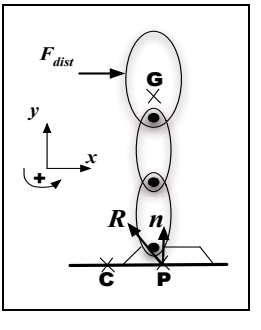


Fig. 3. Model of a humanoid subjected to a horizontal disturbance. Copied from Abdallah *et al.* [3].

ZMP coincides with the CoM. Any increase in F would increase the position of the ZMP with respect to origin causing the robot to become unstable as soon as the ZMP gets closer to the foot edge. Notice that a positive increase in the robot angular momentum can decrease the effect of the disturbance on the ZMP position. Indeed, to damp a disturbance force humans accelerate their torso and rotate their arms creating a rate of angular momentum to compensate for the increase in ZMP [3],[4].

2.2 Momentum Controller

As discussed in Sect. 2 disturbance avoidence is achieved by controlling the angular and linear momenta. Kajita *et al.* [5] express these terms as:

$$\begin{bmatrix} \boldsymbol{L} \\ \boldsymbol{H_G} \end{bmatrix} = \begin{bmatrix} \tilde{m}\boldsymbol{E} & -\tilde{m}\hat{\boldsymbol{r}}_{B\rightarrow\tilde{C}} & \boldsymbol{M}_{\dot{\theta}} \\ \boldsymbol{0} & \tilde{\boldsymbol{I}} & \boldsymbol{H}_{\dot{\theta}} \end{bmatrix} \begin{bmatrix} v_B \\ w_B \\ \dot{\boldsymbol{\theta}} \end{bmatrix}, \tag{3}$$

where $\boldsymbol{E}$ is a 3 x 3 identity matrix, $\tilde{m}$ is the total mass of the robot, $\hat{\boldsymbol{r}}_{B\rightarrow\tilde{C}}$ is the 3 x 1 vector distance from the base (taken at the hips) to the CoM, $\tilde{I}$ is the 3 x 3 inertia matrix with respect to the CoM. $\boldsymbol{M}_{\dot{\theta}}$ and $\boldsymbol{H}_{\dot{\theta}}$ are the 3 x n (n is the number of degrees of frredom) inertia matrices, whose components are calculated as shown by Kajita *et al.* [5] in a recursive manner from one extremity to the body. The final step involves taking the time derivative of (3) and solving for the joint accelerations using the Moore-Penrose pseudo inverse (see [3] for details). Once the accelerations are calculated, one can solve for the joint torques using the dynamic equation of motion [3].

Looking at the operations associated with this solution, one could appreciate the difficulty in implementing them on a small humanoid robot with processing capabilities of a simple microcontroller. For this reason, in the following section we attempt to devise a mush simpler solution requiring less processing, albeit at the price of more stringent constraints and less accuracy.

3 Proposed System

The model developed in this section is simplified from the one shown in Fig 3 by assuming a humanoid with 2 degrees of freedom, one at the hip joint, allowing the upper torso to rotate and a second at the ankle joint allowing the entire body to rotate back. It is further assumed that the weight of the robot is entirely located in the upper torso and the bottom link is massless (see Fig. 2 right).

For any chain model, the angular momentum can be considered to be equal to the momentum of the chain around CoM plus momentum of CoM around origin C.

$$H_y = \sum H_{CoM} + H_C, \tag{4}$$

Given that the mass of the robot is concentrated above the torso and assuming a robot in 2D, one can write 3 as:

$$\begin{aligned} H_C &= I\dot{\theta} + m(\boldsymbol{r}_{CG} \times \boldsymbol{V}), \\ &= I\dot{\theta} + y_{CoM} m V_x + x_{CoM} m V_y, \end{aligned} \tag{5}$$

where $\boldsymbol{r}_{CG}$, $\boldsymbol{V}$, and θ are the position, velocity, and angular velocity respectively of the CoM. Taking the derivative with respect time of (3):

$$\dot{H}_C = I\ddot{\theta} + y_{CoM} m\dot{V}_x + x_{CoM} m\dot{V}_y + \dot{y}_{CoM} m V_x + \dot{x}_{CoM} m V_y. \tag{6}$$

This equation states that five terms affect the change in momentum of the humanoid. We postulate here that by changing only the first term on the right hand side of (3) (*i.e.*, $I\ddot{\theta}$), it is possible to achieve fall avoidance.

For the sake of demonstrating the feasibility of the system a porportional controller is used to control both the hip and ankle joints as follows:

$$\dot{\alpha} = k_1(x_G - x_{G_{init}}), \tag{7}$$

$$\dot{H}_C = k_2(x_{P_{init}} - x_p) \tag{8}$$

where $\dot{\alpha}$ is the rotational velocity of the ankle joint, k_1 and k_1 are proportional gains, and $x_{G_{init}}$ is the initial position of CoM with respect to origin. and $x_{P_{init}}$ is the initial position of ZMP with respect to origin. The instantaneous position of x_g is calculated at each iteration and that of x_P is obtained via force sensor placed on the underside of the feet of the robot. Using (3) and (8) the value of angular acceleration $\ddot{\theta}$ needed to bring back the ZMP to the stable initial position can be computed at each time step.

4 Experiments and Results

To test the proposed algorithm both a simulated and real robot were used. The simulated robot is a 22-DOF humanoid developed under Gazebo [6] called

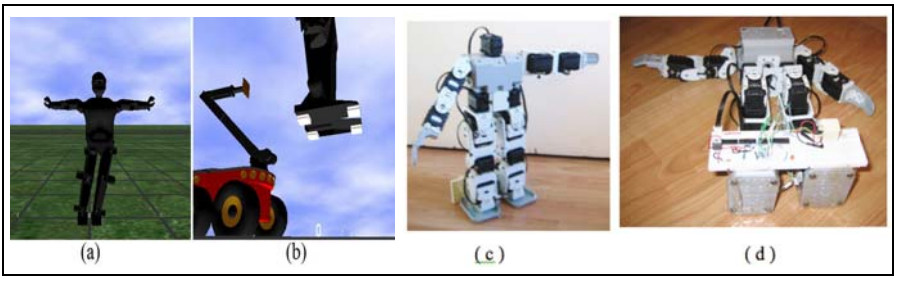


Fig. 4. (a) Baub performing a dance routine in Gazebo, (b) force sensors on the foot soles of Baub, (c) Boloid robot, (d) Boloid showing foot sensors

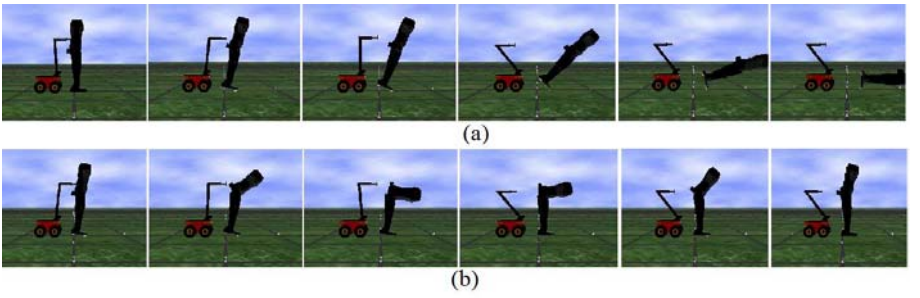


Fig. 5. Response of Baub to a disturbance of 1N. The stiff robot in (a) falls down. The reflex-endowed robot in (b) absorbs the disturbance.

Fig. 6. Response of Bioloid to a disturbance of 1N. The stiff robot in (top) falls down. The reflex-endowed robot in (bottom) absorbs the disturbance.

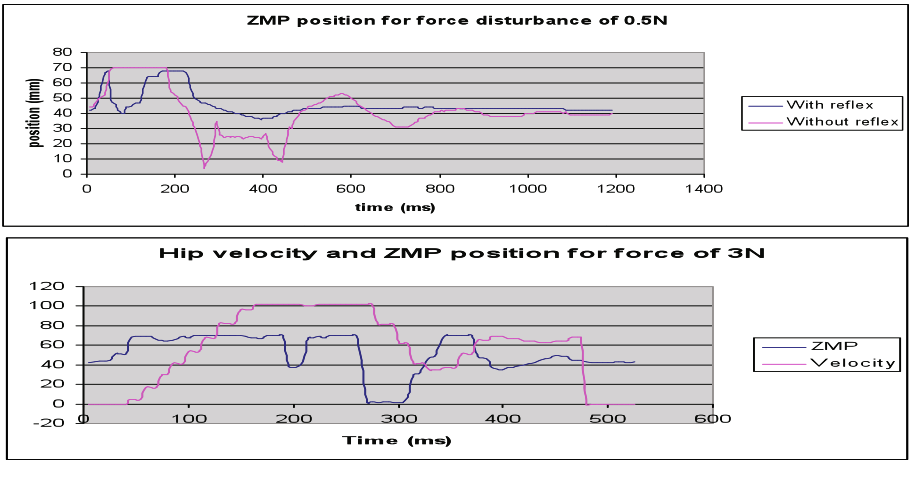


Fig. 7. Response of Bioloid to a disturbance of 1N. The stiff robot in (top) falls down. The reflex-endowed robot in (bottom) absorbs the disturbance.

"Baub" (see Fig. 4-a). The real robot is the Bioloid by CrustCrawler Robotics [2] (see Fig. 4-c)

During the experiments, the head and arms of Baub were removed to reduce the computational burden. The simulations and real test results are respectively shown in Figures 5 and 7 for a push of 1N. In each case a comparison is made between the behavior with and without reflex. Note that for both cases the stiff robot falls down, whereas the reflex endowed robot successfully absorbs the disturbance. Figure

5 Conclusion and Future Work

This paper describes a simple model for a momentum controller for humanoid push recovery. The intent is to provide a system that can be implemented on imbedded systems. Experiments conducted on a simulator show successful disturbance absorption performance. Future work involves implementing fall avoidance in multiple directions as well as implementing a more elaborate control scheme.

References

1. `http://playerstage.sourceforge.net/` (2009)
2. `http://www.crustcrawler.com/products/bioloid/index.php?prod=59` (2009)
3. Abdallah, M., Goswami, A.: A biomechanically motivated two-phase strategy for biped upright balance control. In: Proceedings of the 2005 IEEE International Conference on Robotics and Automation (2005)
4. Azevedo, C., Poignet, P., Espiau, B.: Artificial locomotion control: from human to robots. Robotics and Autonomous Systems 47(4), 203–223 (2004)
5. Kajita, S., Kanehiro, F., Kaneko, K., Fujiwara, K., Harada, K., Yokoi, K., Hirukawa, H.: Resolved momentum control: humanoid motion planning based on the linear and angular momentum. In: Proceedings of the 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems, Las Vegas, Nevada (2003)
6. Koenig, N., Howard, A.: Design and use paradigms for gazebo, an open-source multi-robot simulator. In: Proceedings of the IEEE/RSJ conference on Intelligent Robots and Systems, pp. 2149–2155 (2004)
7. Pratt, J., Carf, J., Drakunov, S., Goswami, A.: Capture point: A step toward humanoid push recovery. In: Proceedings of the IEEE-RAS International Conference on Humanoid Robots, pp. 200–207 (2006)
8. Pratt, J., Goswami, A.: Learning capture points for humanoid push recovery. In: Proceedings of the IEEE-RAS International Conference on Humanoid Robots (2007)
9. Sardain, P., Bessonnet, G.: Forces acting on a biped robot. center of pressure-zero moment point. IEEE transactions on Systems, Man, and Cybernetics part A: Systems and Humans, SMC 2004 34(5), 630–637 (2004)
10. Vukobratović, M., Borovac, B., Raković, M., Potkonjak, V., Milinović, M.: On some aspects of humanoid robots gait synthesis and control at small disturbances. International Journal of Humanoid Robotics 5(1), 119–156 (2008)

Optimal Trajectory Generation for Walking Up and Down a Staircase with a Biped Robot Using Genetic Algorithm (GA)

Eunsu Kim, Manseok Kim, and Jong-Wook Kim

Department of Electronic Engineering, Dong-A Universisy,
840 Hadan2dong, Busan, Korea
tacctics@hotmail.com, luhyoun@hotmail.com, kjwook@dau.ac.kr

Abstract. In this paper, a humanoid is simulated and implemented to walk up and down a staircase using the blending polynomial and genetic algorithm (GA). Both ascending and descending a staircase are scheduled by four steps. Each step mimics natural gait of human being and is easy to analyze and implement. Optimal trajectories of ten motors in a lower extremity of a humanoid are rigorously computed to simultaneously satisfy stability condition, walking constraints, and energy efficiency requirements. As an optimization method, GA is applied to search optimal trajectory parameters in blending polynomials. The feasibility of this approach will be validated by simulation with a small humanoid robot.

Keywords: Humanoid robot, Heuristic algorithm, Biped robot, Genetic algorithm.

1 Introduction

Human being's behavior is very efficient and stable. Among the behaviors, they walk stably under various environments with the pattern that uses minimal energy. Since humanoids should be able to accompany humans for intelligent services wherever they go, walking up and down on a staircase is required as basic mobility.

In walking upstairs and downstairs with a biped robot, generation of trajectories for all the joint motors is very difficult and important. For stable upstairs walking, the tip of the foot in the swaying leg must not stub against the stair edge, the foot should be landed on the upper stair safely. And for stable downstairs walking, the heel of the foot in the swaying leg must not stub against the stair edge, the foot should be landed on the lower stair safely, too. In doing so, the zero moment point (ZMP) should be within the convex hull of contact points between the feet and the ground. Moreover, to cope with a situation of using battery, least energy consumption by minimizing sum of motor torques should also be considered in trajectory planning [1]. These multiple objectives can be effectively achieved by using a computational optimization

J.-H. Kim et al. (Eds.): FIRA 2009, LNCS 5744, pp. 103–111, 2009.

method also known as metaheuristics. Real-coded genetic algorithm (RCGA) was successfully applied to optimal trajectory generation of biped robot [2].

In this paper, adaptive GA (AGA) [3] is applied to optimal trajectory generation for walking up and down a staircase with a biped robot that has in total eight rigid links, ten joint motors, with seven degrees of freedom (DOF). To generate the ten joint motor trajectories, blending polynomials composed of two or three segments are employed [4]. In order to raise search efficiency, assumptions on robot posture while walking are also applied [5].

In this paper, both ascending and descending a staircase are scheduled by four steps. Each step mimics natural gait of human being and is easy to analyze and implement. The optimal trajectory parameters attained by AGA are applied to walking simulation, and the resulting trajectories will be quantized and commanded to joint motors in a small humanoid robot for validation of feasibility.

2 Dynamic Model of a Biped Robot

2.1 Humanoid Model

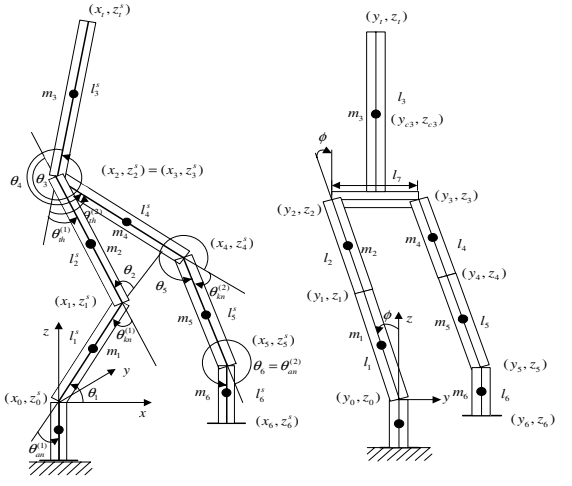
Fig. 1 shows the present biped robot models from the viewpoint of sagittal and coronal planes specified in the Cartesian coordinate system. In the figure, l_i and m_i denote the length and mass of the link i, respectively and two types of joint angles are employed as coordinates q; Denavit-Hartenberg (DH) angles $\theta_i, i = 1,\ldots,2$ and actual motor angles $\theta_{kn}^{(i)}, \theta_{kn}^{(i)}, \theta_{kn}^{(i)}, i = 1,2$.

Geometric observation leads to following relation as

$$\begin{aligned} &\theta_1(t) = \frac{\pi}{2} - \theta_{an}^{(1)}(t), \quad \theta_2(t) = \theta_{kn}^{(1)}(t), \quad \theta_3(t) = -\theta_{th}^{(1)}(t), \\ &\theta_4(t) = \pi + \theta_{th}^{(2)}(t), \quad \theta_5(t) = -\theta_{kn}^{(2)}(t), \quad \theta_6(t) = \theta_{an}^{(2)}(t). \end{aligned} \tag{1}$$

For walking with a biped robot, ankle joints are rotated with ϕ° degrees in the coronal plane, and with θ_i° degrees in the sagittal plane simultaneously. As a combination of these independent movements, the lengths of each link are projected in sagittal plane as

$$\begin{aligned} &l_1^s = l_1 \cos\phi(t), l_2^s = l_2 \cos\phi(t), l_3^s = l_3, \\ &l_4^s = l_4 \cos\phi(t), l_5^s = l_5 \cos\phi(t), l_6^s = l_6 \end{aligned} \tag{2}$$



(a) sagittal plane (b) coronal plane

Fig. 1. Modeling of a biped robot

By using the concept of projection, three-dimensional coordinates of all the joints are to be easily derived in both planes. The coordinates of six joints, essential for computing kinematics and ZMP, are actually as simple as shown below:

$$
\begin{aligned}
x_1 = x_0 + l_1^S C_1, y_1 = y_0 - l_1^C S_\phi, z_1 = z_0 + l_1^S S_1, \\
x_2 = x_1 + l_2^S C_{12}, y_2 = y_1 - l_2^C S_\phi, z_2 = z_1 + l_2^S S_{12}, \\
x_3 = x_2, y_1 = y_2 - l_7, z_3 = z_2, \\
x_4 = x_3 + l_4^S C_{1234}, y_4 = y_3 - l_4^C S_\phi, z_4 = z_3 + l_4^S S_{1234}, \\
x_5 = x_4 + l_5^S C_{12345}, y_5 = y_4 - l_5^C S_\phi, z_5 = z_4 + l_5^S S_{12345}, \\
x_6 = x_5 + l_6^S C_{123456}, y_6 = y_5, z_6 = z_5 + l_6^S S_{123456}
\end{aligned}
\tag{3}
$$

where $C_{12\cdots j}$, $S_{12\cdots j}$, and S_ϕ represent $\cos(\theta_1+\theta_2+\cdots+\theta_j)$, $\sin(\theta_1+\theta_2+\cdots+\theta_j)$, and $\sin\phi$ respectively. In the case of supporting with the right leg, only y_3 is changed in (3) as $y_3 = y_2 + l_7$.

Dynamic equation of biped robot is described with the Euler-Lagrange equation as

$$
\sum_{j=1}^{6} d_{kj}(q)\ddot{q}_j + \sum_{i=1}^{6}\sum_{j=1}^{6} c_{ijk}\dot{q}_i\dot{q}_j + g_k(q) = \tau_k, k = 1,\cdots,6, \tag{4}
$$

where q_i denotes the i th generalized coordinate, τ_k the k th joint torque, d_{ij} the entries of the 6 x 6 inertia matrix $D(q)$, c_{ijk} the Cristoffel symbol such that

$$c_{ijk} = \frac{1}{2}\left\{\frac{\partial d_{kj}}{\partial q_i} + \frac{\partial d_{ki}}{\partial q_j} - \frac{\partial d_{ij}}{\partial q_k}\right\} \tag{5}$$

In this paper, trajectories of the joint motor angles are generated using adaptive GA, and blending polynomials are used to approximate joint trajectories [5]. Cubic B-spline is the smoothest function which provides minimal basis for fixed dimension space and thus was already applied to trajectory planning of a biped robot [1].

The blending polynomial generates a trajectory connected by trajectory segments of low order polynomials between adjacent via points. With this approach, care must be taken that velocity and acceleration constraints are satisfied at via points.

The required cubic polynomial for this segment of the trajectory can be computed from the following equation:

$$\begin{aligned} &q(t) = a_0 + a_1(t-t_0) + a_2(t-t_0)^2 + a_3(t-t_0)^3 \\ &a_0 = q_0, a_1 = v_0, \\ &a_2 = \frac{3(q_1-q_0) - (2v_0+v_1)(t_f-t_0)}{(t_f-t_0)^2}, \\ &a_3 = \frac{3(q_0-q_1) + (v_0+v_1)(t_f-t_0)}{(t_f-t_0)^3}. \end{aligned} \tag{6}$$

where the initial/final angles/velocities are defined as $q(t_0) = q_0, \quad q(t_f) = q_1, \quad \dot{q}(t_0) = v_0, \quad \dot{q}(t_f) = v_1$.

In walking simulation, the three time parameters t_0, t_m, t_f, and the two velocity parameters v_0, v_f, and the two angle parameters q_0 and/or q_f in Eq. (6) are presumed as fixed values for smooth walking. Therefore the actual number of parameters to be optimized is maximally three in each trajectory. Details are explained in the following section.

2.2 Walking Up a Staircase

Human beings walk upstairs by first lifting up a swaying leg while maintaining standing stability, landing it on the upper stair, moving ZMP to the landing foot, repeating this process for the supporting leg left behind, and finally standing upright at the upper stair. As the first approach, we deal with the case a robot is walking up only one stair at each gait.

At the first step, the foot of a swaying leg is lifted up higher than the height of a stair not to touch its edge by using inverse kinematics. In an initial posture, in order to stabilize the robot body, the ZMP is moved inside the foot area of the supporting leg. It is easily derived as

$$(x_6, y_6, z_6) = (x_0, y_0 - l_7, 0). \tag{7}$$

The coordinate of the lifted leg up to h_f using Eq. (7) is derived as

$$(x_6', y_6', z_6') = (x_0, y_0 - l_7, h_f). \tag{8}$$

For the lifted foot to be vertical to ground, the following condition should be hold:

$$\theta_1 + \theta_2 + \theta_3 + \theta_4 + \theta_5 + \theta_6 = -\frac{\pi}{2}, \tag{9}$$

where the angle of supporting leg which is not changed in the sagittal plane as a standing position are defined as $\theta_1 = \frac{\pi}{2}, \theta_2 = \theta_3 = 0$. By using this condition and Eq. (3), solving Eq. (8) and (9) yields the following equations:

$$\begin{aligned} x_6' &= x_0 + l_4^s \cos\left(\theta_{th}^r - \frac{\pi}{2}\right) + l_5^s \cos\left(\theta_{th}^r - \theta_{kn}^r - \frac{\pi}{2}\right) = x_0 \\ y_6' &= y_0 - l_7 - l_1 S_\phi - l_2 S_\phi - l_4 S_\phi - l_5 S_\phi = y_0 - l_7 \\ z_6' &= z_3 + l_4^s \sin\left(\frac{\pi}{2} + \theta_4\right) + l_5^s \sin\left(\frac{\pi}{2} + \theta_4 + \theta_5\right) + l_6^s \sin\left(-\frac{\pi}{2}\right) = h_f \end{aligned} \tag{10}$$

and the DH angles in Eq. (10) are changed into their motor angles, which yields the following relations:

$$\begin{aligned} & l_4^s \cos\left(\theta_{th}^r - \frac{\pi}{2}\right) + l_5^s \cos\left(\theta_{th}^r - \theta_{kn}^r - \frac{\pi}{2}\right) = 0 \\ & S_\phi \left\{ l_1 \left(1 + \sin\left(\theta_{th}^r - \theta_{kn}^r - \frac{\pi}{2}\right)\right) + l_2 \left(1 + \sin\left(\theta_{th}^r - \frac{\pi}{2}\right)\right) \right\} = 0 \\ & l_1^s + l_2^s + l_4^s \sin\left(\theta_{th}^r - \frac{\pi}{2}\right) + l_5^s \sin\left(\theta_{th}^r - \theta_{kn}^r - \frac{\pi}{2}\right) - h_f = 0 \end{aligned} \tag{11}$$

This motion requires proper $\theta_4, \theta_5, \theta_6$ that can be attained for a desired h_f by AGA. In lifting up a foot, energy consumption is not of interest. Thus the two actuating motors rotate linearly from zero to the target angle.

At the second step, the lifted foot is put forward and landed at the face of upper stair. During this motion, ZMP should be located inside the foot of the supporting leg, and for the least energy consumption each joint should rotate in a smooth way which is scheduled by blending polynomials and AGA. The central position of the landed foot is set at the center of the stair face.

At the third step, the landed foot is ready to support the body by shifting ZMP from the behind leg to the forward one. By using inverse kinematics, the final thigh angle of the supporting leg after moving ZMP is easily derived as

$$\theta_4 = \left(\cos^{-1} \left(\frac{h_f^2 - 2l_5^2}{-2l_5^2} \right) - \pi \right) \tag{12}$$

And, the final ankle angle of the leg left behind is

$$\theta_6 = \left(\cos^{-1} \frac{\sqrt{l_{ft_{for}}^2 + l_6^2}}{\sqrt{l_{ft_{for}}^2 + l_6^2} - l_{cor}} \right) - \theta_4 \tag{13}$$

And by using the first step, the leg angle of the upper stair is easily derived.

As a result, the upper body is translated forward and the hill of the foot in the lower stair is sufficiently lifted to maintain the current double support phase. Like the first step, each joint rotates linearly from the angles reached after the second step to the appropriate angles.

At the final step, the leg left behind is moved up to make the posture of standing upright. This finalizes a period of ascending one stair, and the next ascending starts from the first step above mentioned until the robot ascends all the stairs. While lifting up the leg and landing its foot, the toes of the foot should not collide with the stair edge and the sole of it should not below the stair face before landing. These requirements are reflected in penalty terms added to the cost function minimized by AGA.

2.3 Walking Down a Staircase

Descending a staircase is also scheduled by four steps; at the first step, the foot of swaying leg is put forward to the empty space. During this motion, ZMP should be located inside the foot of the supporting leg, and for the least energy consumption each joint should rotate in a smooth way which is scheduled by blending polynomials and AGA.

At the second step, the knee of the supporting leg is bent lower as much as the stair height for landing the swaying foot on the lower stair. During this motion, ZMP should be located inside the foot of supporting leg, since the position of the supporting leg is lower. Like the first step of walking up a staircase, each joint rotate linearly from the angles reached after the first step to the appropriate angles.

At the third step, the landed foot on the lower stair is ready to support the body by shifting ZMP from the leg behind, and it should be stretched. By using inverse kinematics, the final thigh angle of the supporting leg after moving ZMP is derived as

$$\theta_4 = \tan^{-1} \left(\frac{l_1 + l_2 - h_f}{l_{ft}} \right) - \cos^{-1} \left(\frac{0.5\sqrt{(l_1 + l_2 - h_f)^2 + (l_{ft})^2}}{l_4} \right) \tag{14}$$

And, the final ankle angle of the leg on upstairs behind is

$$\theta_6 = \frac{\pi}{2} - \tan^{-1} \left(\frac{l_1 + l_2 - h_f}{l_{ft}} \right) - \cos^{-1} \left(\frac{0.5\sqrt{(l_1 + l_2 - h_f)^2 + (l_{ft})^2}}{l_5} \right) \tag{15}$$

The final knee angle of the upstairs leg is derived as

$$\theta_5 = \pi - (\theta_4 + \theta_6) \tag{16}$$

As a result, the upper body is translated forward and the leg of downstairs is stretched. Like the second step, each joint rotates linearly from the angles reached after the second step to the appropriate angles.

At the final step, the leg of upstairs is moved down to make the posture of standing upright. This finalizes a period of descending a stair, and the next descending is carried out by iterating the above four steps until the robot descends all the stairs. These requirements are reflected in penalty terms added to the cost function minimized by AGA as walking up a staircase.

3 Simulation of Walking Upstairs

3.1 Problem Formulation

Mechanical efficiency is crucial to the viability of any mobile robot, since it determines the range of useful movement for a given amount of fuel. Thus the cost function to be minimized with AGA in this paper is defined as

$$J(X) = \gamma_C T_S \sum_{m=1}^{N} \tau(m)^T \tau(m) + P(X), \boldsymbol{\tau}(m) = \left[\tau_1(m)\ \tau_2(m)\ \tau_3(m)\ \tau_4(m)\ \tau_5(m)\ \tau_6(m)\right]^T \tag{17}$$

where γ_C is the weighting constant for the energy cost in multi-objective optimization, N is the number of sample points, τ is the joint torque computed by Eq. (3).

Since AGA is inherently an unconstrained optimization problem, the exterior penalty function method is useful for describing all the constraints using the maximum and minimum functions like the following

$$\begin{aligned} P(X) = {} & \gamma_1 \frac{\left|x_{tip}(T) - S\right|}{S} + \gamma_2 \frac{\left|z_{tip}(T) - h_f\right|}{h_f} + \gamma_3 \left(\text{violation of swaing leg's minimum height}\right) \\ & + \gamma_4 \left(\text{violation of joint limit angles}\right) + \gamma_5 \left(\text{violation of ZMP stability}\right) \end{aligned} \tag{18}$$

where γ_i are the penalty parameters chosen to keep from violating the constraints, and h_f which is not considered now is the desired height of a staircase. S and h_f are used for normalization of the associated costs.

In conclusion, the problem considered in this paper can be stated as follows: Given dynamical Eq. (3), find all the joint trajectories that minimize Eq. (17) subject to the constraints of Eq. (18). This complicated task is simply transformed to an unconstrained optimization problem by minimizing an augmented cost $J + P$.

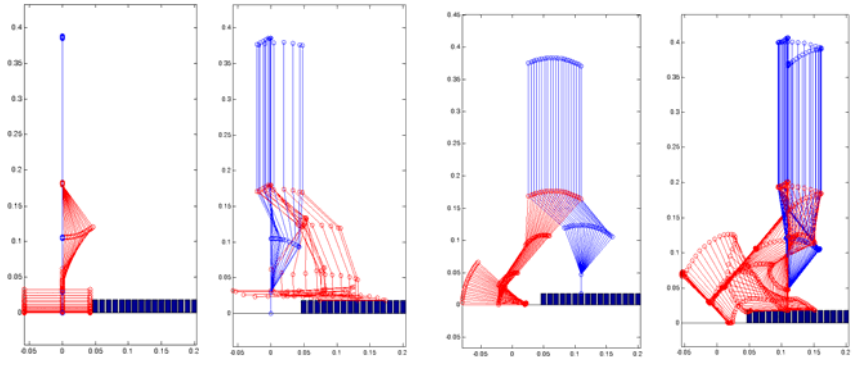
3.2 Problem Formulation

Our simulation code calculates initial and final postures of the four-step segments for walking up and down a stair, and it also generates smooth trajectories using the blending polynomial method. The generated trajectories are converted to physical motor angles for direct application to a humanoid kit.

Simulation condition on walking upstairs and downstairs is selected as

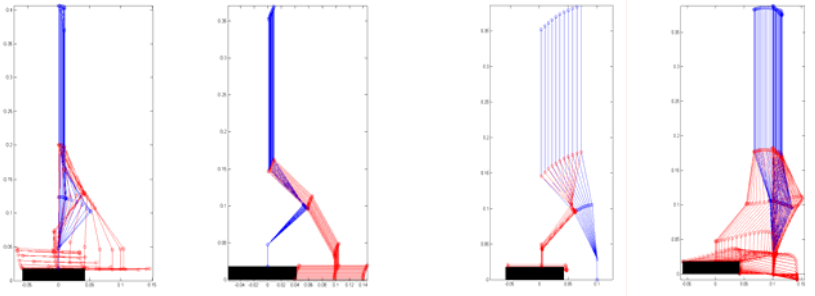
- Stride length (S) = 0.1m
- Walking Upstairs
  - First step period = 1.5 sec
  - Second step period = 1.0 sec
  - Third step period = 0.5 sec
  - Final step period = 1.5 sec
- Walking Downstairs
  - First step period = 1.0 sec
  - Second step period = 1.5 sec
  - Third step period = 0.5 sec
  - Final step period = 1.5 sec
- Staircase height (h_f) = 0.018m

Fig. 2 shows stepwise processes of walking upstairs implemented in simulation, and Fig. 3 shows downstairs walking.



(a) first step (b) second step (c) third step (d) final step

Fig. 2. Simulation of walking up a staircase separated by the proposed four steps



(a) first step (b) second step (c) third step (d) final step

Fig. 3. Simulation of walking down a staircase separated by the proposed four steps

4 Conclusion

This paper proposes a new approach of trajectory generation for humanoid's walking upstairs and downstairs using blending polynomial and AGA. The patterns for walking upstairs and downstairs are divided into four steps for analysis and implementation, and the optimal trajectories of ten motors in the lower body are generated using AGA. Owing to the effectiveness of describing trajectories with blending polynomial and the good optimization capability of AGA, the proposed approach is validated feasible via simulation. As future work, the proposed approach will be implemented with a humanoid kit.

Acknowledgments. This work was supported by the Grant of the Korean Ministry of Education, Science and Technology" (The Regional Core Research Program/Institute of Logistics Information Technology).

References

1. Lee, T.-T., Chen, Y.-C.: Minimum-fuel path planning of a 5-link biped robot. In: Proc. of the Twentieth Southeastern Symp. on System Theory, pp. 459–463 (1988)
2. Jeon, K.S., Kwon, O., Park, J.H.: Optimal trajectory generation for a biped robot walking a staircase based on genetic algorithm. In: Proc. International Conference on Intelligent Robots and Systems, pp. 2837–2842 (2004)
3. Srinivas, M., Patnaik, L.M.: Adaptive probabilities of crossover and mutation in genetic algorithms. IEEE Trans. on Systems, Man and Cybernetics 24(4), 659–667 (1994)
4. Spong, M.W., Hutchinson, S., Vidyasagar, M.: Robot Modeling and Control. John Wiley & Sons, Inc., Chichester (2006)
5. Kim, T.G., Kim, J.-W.: Planning walking patterns of a biped robot with uDEAS optimization. In: International Conference on Control, Automation and Systems, pp. 2693–2698 (2007)

A Study on the Motion Energy of Biped Robot Walking on Different Postures

Kuo-Yang Tu* and Mi-Shin Liu

Institute of System Information and Control,
National Kaohsiung First University of Science and Technology
2 Juoyue Rd., Nantsu District, Kaohsiung 811, Taiwan, R.O.C.
tuky@ccms.nkfust.edu.tw

Abstract. It is interested for a bipedal robot manipulated by different Degree of Freedoms (DoFs) to spend how much motion energy. In this paper, the motion energy of a biped robot influenced by the manipulation of different DoFs is studied. For calculating the motion energy, the forward and inverse kinematics of the designed biped robot are first derived. The 4-3-4 trajectory for planning the movement of robot joints in smoothing is designed for the walking finished by the biped joint movement. Once the joint trajectories of the biped robot are solved, the motion energy including kinetic and potential energy can be calculated. The walking of the biped robot manipulated by 2, 4 and 6 DoFs, respectively, is also included. The study provides the mechanical design of biped robots in the future.

Keywords: Biped Robot; Walking Pattern; Degree of Freedoms; Motion Energy.

1 Introduction

It is interested for a bipedal robot manipulated by different Degree of Freedoms (DoFs) to spend how much motion energy. It is concerned with mechanical design, and DoFs used for walking and planning. The effectiveness of motion energy determines the walking period of a biped robot in individually executing a task. It is important for biped robots in the further developments.

The walking of biped robots includes the basic works such as single support phase, double support phase, Center Of Gravity (COG), Zero Moment Point (ZMP), and so on. The planning of a swing leg put on an appropriate location is the basic work of biped walking. For biped walking like human, Harada et al. proposed the analytical method for real-time planning [1]. Based on the trajectories of previous planning, the accident event leads modified trajectories built by the biped robot ZMP and COG. Asano et al. proposed the idea of the walking patterns produced by dynamic [2]. From the point of view in mechanical systems, energy change concerns with varying COG. This point of view formulates walking patterns into partial differential equations

* This research was partly supported by National Science Council, Taiwan, Rep. of China under grant NSC 95-2213-E-327-015.

J.-H. Kim et al. (Eds.): FIRA 2009, LNCS 5744, pp. 112–124, 2009.

solved for the planning of biped robot. Agrawal designed the new motion control of biped walking in planar [3]. The biped dynamics approximated to linear system design legs as individual inertial matrices according to COG. The nonlinear terms including centrifugal and coriolis force is ignored. Its dynamics become linear and easy to been controlled. Shih designed the biped robot for going up and down stair [4]. The designed walking patterns based on kinematics, dynamics and impact factors are supported by tough sole providing friction force enough. In special, the torso balance is performed by a slid literal body. Huang designed reflective control for the planned walking patterns [5, 6]. According to the constraints of actuators, joint torque and velocity, planning trajectories stabilizes the biped robot in sequent walking. The planning trajectories include soles and waist.

In hits paper, the walking motion energy of the biped robot manipulated by different DoFs is studied. It is surprised that the biped robot manipulated by two DoFs is unsmooth walking like a constrained action, but spends less motion energy.

The rest of this paper is organized as follows. The section 2 presents the design and model of a biped robot used to study the motion energy influenced by the manipulation of DoFs. In section 3, the strategy for the planning of continuous joint trajectory is proposed. In section 4, the inverse kinematics of the biped robot designed for this research is solved. Section 5 discusses the idea of the biped robot manipulated by different DoFs. In section 6, the motion energy of the biped robot is formulated. The simulation for calculating the motion energy in different DoFs is also included. Finally, Section 7 is conclusion.

2 Design and Model of a Biped Robot

Fig. 1 is the mechanical diagram of the designed of the biped robot. As shown in Fig. 1, the joint coordinate is also defined. It consists of twelve links including thigh, calf, knee and ankle. The biped robot consisted of twelve joints to connect its links. The biped robot is thus twelve DoFs. Table 1 shows the specification of the biped robot. This Table also depicts a virtual torso. Length 100 mm, weight 3 Kg, inertia torque 0.02 Kgm2. The virtual torso lets the study of the biped motion energy much real.

The coordinate system of every joint of the biped robot is defined based on D-H rule. According to the coordinate system and the link parameters, the forward kinematics of the biped robot can be derived. The forward kinematics can solve the pose (the position and orientation of every link) of the biped robot from joint angles.

Table 1. The parameters of the biped robot

Link	1, 12	2, 11	3, 10	4, 9	5, 8	6, 7
Length (mm)	X=30 Z=25	Z=25	Y=20 Z=130	Z=85	Z=40	Z=30
Weight (kg)	0.8	0.5	1.5	1	0.7	0.6
Inertia(kgm^2)	0.005	0.003	0.01	0.006	0.0047	0.004

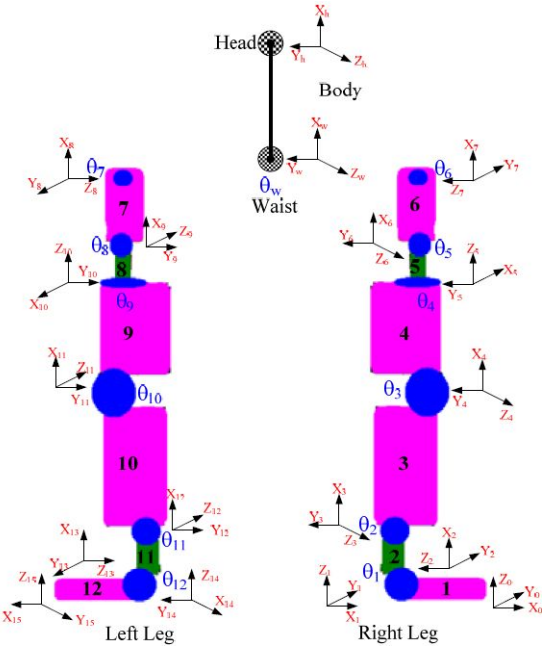


Fig. 1. The mechanical diagram and joint coordinate system of the designed biped robot

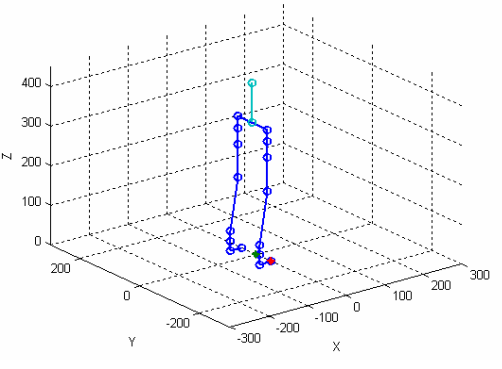


Fig. 2. The biped robot in right support phase

In right support phase, the pose of the biped robot can be figured out by its forward kinematics. The derived forward kinematics calculates every link position and orientation of the biped robot in right support phase as shown in Fig. 2. Fig. 2 also demonstrates that the derived kinematics is correct. Since the biped robot is symmetric, the pose of left support can be obtained as well.

3 Trajectory Planning of the Biped Robot

Continuity in time is the basic need of trajectory planning of biped robots. There are many polynomials used for the trajectory planning such as three cubic, five cubic, and 4-3-4, etc. In this study, the 4-3-4 polynomial for the trajectory planning of the biped robot is proposed. The 4-3-4 polynomial has three functions of time as follows

$$h_1(t) = a_{14}t^4 + a_{13}t^3 + a_{12}t^2 + a_{11}t + a_{10} \tag{1}$$

$$h_2(t) = a_{23}t^3 + a_{22}t^2 + a_{21}t + a_{20} \tag{2}$$

$$h_f(t) = a_{f4}t^4 + a_{f3}t^3 + a_{f2}t^2 + a_{f1}t + a_{f0} \tag{3}$$

The three polynomials pass four locations of joint angles. Let the four locations be (t_0, θ_0), (t_1, θ_1), (t_2, θ_2), and (t_f, θ_f), the initial angular velocity and acceleration are ω_0, and α_0, respectively, and the final angular velocity and acceleration are ω_f and α_f, respectively. In total, the three polynomials need fourteen boundary conditions. In addition to the above eight conditions, the rest six conditions is the constraints of the polynomials continuous at lift-up location θ_1 and set-down location θ_2. A numerical example as $\theta_0 = 0°$, $\theta_1 = 20°$, $\theta_2 = 75°$, $\theta_f = 90°$, $\omega_0 = 0$, $\alpha_0 = 0$, $\omega_f = 0$, and $\alpha_f = 0$ between the time step $t_0 = 0$, $t_1 = 2$, $t_2 = 7$, $t_f = 10$ is shown in Fig. 3.

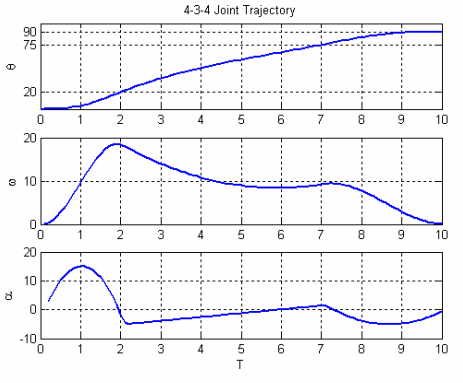


Fig. 3. The position, velocity and acceleration of the 4-3-4 trajectory

It is not enough to just use one 4-3-4 polynomial for a smooth joint trajectory of the biped robot. Every one of joint trajectory consists of two 4-3-4 polynomials. Tow 4-3-4 polynomials have seven interpolating points to assign the joint angles. The previous four points consist of the first polynomial, and the posterior four points consist of the second polynomial.

Let the seven locations be (t_0, θ_0), (t_1, θ_1), (t_2, θ_2), (t_3, θ_3), (t_4, θ_4), (t_5, θ_5), and (t_f, θ_f). And the velocity and acceleration we can assign are ω_0, ω_h, ω_f, α_0, α_h, and α_f. A numerical example as $\theta_0 = 0°$, $\theta_1 = 20°$, $\theta_2 = 75°$, $\theta_3 = 80°$, $\theta_4 = 60°$, $\theta_5 = 15°$, $\theta_f = 0°$, $\omega_0 = 0$, $\alpha_0 = 0$, $\omega_h = 0$, $\alpha_h = 0$, $\omega_f = 0$, $\alpha_f = 0$, $t_1 = 1$, $t_2 = 2$, $t_3 = 3$, $t_4 = 4$, $t_5 = 5$, and $t_f = 6$ is shown in Fig. 4. The biped robot makes use of the trajectory to build its walking patterns.

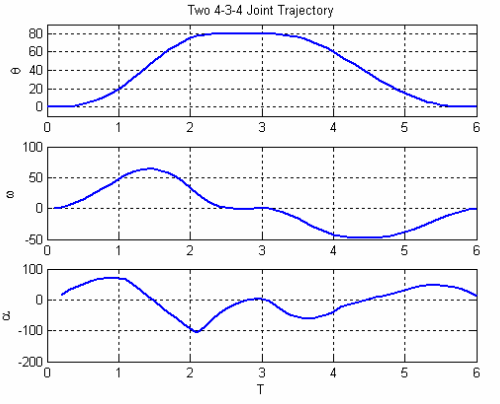


Fig. 4. The trajectory of two 4-3-4 polynomials

4 The Inverse Kinematics of the Joint Trajectories for Appropriate Pose

Exactly the joint trajectories are solved by inverse kinematics. In this section, the inverse kinematics of the biped robot is thus derived.

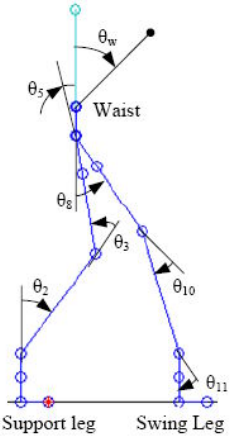


Fig. 5. The concern joints of the biped robot in right support phase

There are twelve joints in the biped robot. In sagittal plane, only six joints concern with the move of the biped robot. Fig. 5 shows the biped robot during the right support phase in sagittal plane. As Shown in Fig. 5, the joints concerned with the biped robot in sagittal plane are of θ_2, θ_3, θ_5, θ_8, θ_{10} and θ_{11}.

If the trajectory planning of the biped robot joints only makes use of two points under the soles of the support leg and the swing leg, respectively, it is possible to have has unreasonable waist that results in inclined torso. Inclined torso makes the control

problem of the biped robot more difficult. One reasonable solution for the joint trajectories of the biped robot is the inverse kinematics separated by three parts that the support leg is fixed at a point, the swing leg moves on a planning trajectory, and the waist follows a planning trajectory. Therefore, three points solve the joint angles with the inverse kinematics in every time step.

Solving the inverse kinematics can be separated into two pats:

I. From support leg to Waist:

$$ {}^{7}_{0}T = \begin{bmatrix} (1) & 0 & (2) & (5) \\ 0 & 1 & 0 & 0 \\ (3) & 0 & (4) & (6) \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & 0 & -1 & X_W \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & Z_W \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{4} $$

II. From waist to swing leg:

$$ {}^{14}_{7}T = \begin{bmatrix} (7) & 0 & (8) & (11) \\ 0 & 1 & 0 & 100 \\ (9) & 0 & (10) & (12) \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & X_F \\ 0 & 1 & 0 & 100 \\ 0 & 0 & 1 & Z_F \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5} $$

Finally, the Eqs. (4) and (5) can solve the inverse kinematics as follows:

$$ \theta_2 = \mathrm{A}\tan 2\left(S_2 / C_2\right) = \mathrm{A}\tan 2\left(\left(\pm\sqrt{1-C_2^2}\right) \Big/ \left(\frac{-2\mathrm{NP} \pm \sqrt{4\mathrm{N}^2\mathrm{P}^2 - 4\left(\mathrm{M}^2+\mathrm{N}^2\right)\left(\mathrm{P}^2-\mathrm{M}^2\right)}}{2\left(\mathrm{M}^2+\mathrm{N}^2\right)} \right) \right) \tag{6} $$

$$ \theta_3 = A\tan 2\left(S_3 / C_3\right) = A\tan 2\left(\left(C_2 Q - S_2 R\right) \Big/ \left(S_2 Q + C_2 R\right) \right) \tag{7} $$

$$ \theta_5 = A\tan 2\left(S_5 / C_5\right) = A\tan 2\left(-S_{23} / C_{23}\right) \tag{8} $$

$$ \theta_8 = A\tan 2\left(S_8 / C_8\right) = A\tan 2\left(\left(\pm\sqrt{1-C_8^2}\right) \Big/ \left(\frac{-2VW \pm \sqrt{4V^2W^2 - 4\left(U^2+V^2\right)\left(W^2-U^2\right)}}{2\left(U^2+V^2\right)} \right) \right) \tag{9} $$

$$ \theta_{10} = A\tan 2\left(S_{10} / C_{10}\right) = A\tan 2\left(\left(C_8 L - S_8 K\right) \Big/ \left(S_8 L + C_8 K\right) \right) \tag{10} $$

$$ \theta_{11} = A\tan 2\left(S_{11} / C_{11}\right) = A\tan 2\left(-S_{810} / C_{810}\right) \tag{11} $$

Eqs. (6) – (11) are the inverse kinematics of the biped robot in right support phase. Since the structure of the biped robot is symmetric, the inverse kinematics of left support phase can be easily derived from that of right support phase.

5 Planning of the Walking Patterns in Distinct DOFs

In the study, the motion energy of the biped robot manipulated by different DoFs is analyzed and compared. The comparison includes the motion energy of two, four and six DoFs.

In the study, the planning of the biped robot walking includes three types: two, four and six DoFs. The manipulation of two DoFs makes use of both support leg's ankle and swing leg's thigh. As shown in Fig. 6, the biped robot manipulated by two DoFs does not have knee. The biped robot manipulated by the ankle and knee of support leg and the thigh and knee of swing leg is of four DoFs as shown in Fig. 7. As shown in Fig. 8, six DoFs include the ankle, knee and thigh of both support and swing leg. The manipulation of the biped robot DoFs is related to the structure of its mechanism. This study provides the idea to design the mechanism of a biped robot.

The manipulation of two DoFs is the simplest walking of the biped robot. The ankle supports the robot, and the thigh swing the leg for a walking stride. However, this manipulation limits the walking type of the biped robot.

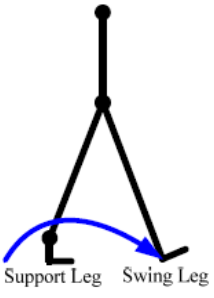


Fig. 6. The manipulation of the biped robot in two DoFs

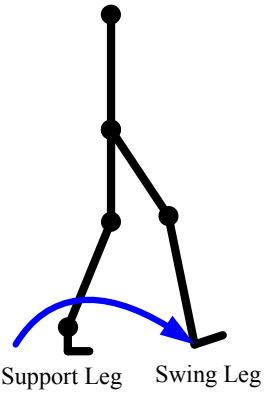


Fig. 7. The manipulation of the biped robot in four DoFs

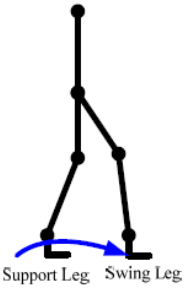


Fig. 8. The manipulation of the biped robot in six DoFs

The 2-DoFs manipulation operates joint (θ_2) and Joint 5 (θ_8) as the biped in right support phase, and operates joint 11 (θ_{11}) and joint 5 (θ_5). Besides, the rest joints of the biped robot are fixed. Suppose a stride of the 2-DOFs manipulation is 270 mm. Let the set points of θ_2 be 0°, 0°, 0°, 0°, -1°, -5°, -12°, and those of θ_8 be 30°, 20°, 10°, 0°, -10°, -20°, -30° as the biped in right support phase, and those of θ_5 be 30°, 20°, 10°, 0°, -10°, -20°, -30°, and those of θ_{11} be 0°, 0°, 0°, 0°, -1°, -5°, -12° in left support phase. The two 4-3-4 polynomials and forward kinematics can be used to calculate the soles of both right and left legs as shown in Fig. 9.

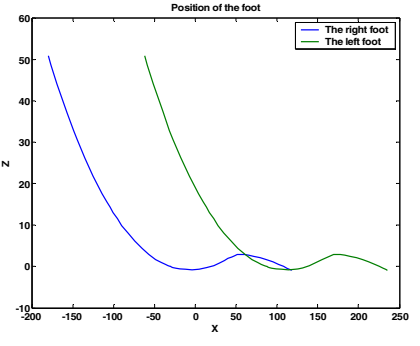


Fig. 9. The trajectories of both right and left soles as manipulating the biped robot in two DoFs

Adding the operation of knee in the 2-DoFs manipulation is four DoFs. Thus, θ_2, θ_3, θ_8 and θ_{10} are operated in right support phase, but θ_3, $\theta 5$, θ_{10} and θ_{11} are operated in left support phase.

Suppose a stride of the 4-DoFs manipulation is 300 mm. Let the set points be θ_2 = 5°, 0°, 0°, 0°, -5°, -12°, -24°, θ_3 = 25°, 10°, 0°, 0°, 0°, 10°, 30°, θ_8 = -15°, -5°, 8°, 15°, 20°, 25°, 25°, θ_{10} = -40°, -40°, -40°, -40°, -35°, -30°, -10° in right support phase, and be θ_3 = 40°, 40°, 40°, 40°, 35°, 30°, 10°, θ_5 = 15°, 5°, -8°, -15°, -20°, -25°, -25°, θ_{10} = -25°, -10°, 0°, 0°, 0°, -10°, -30°, θ_{11} = -50°, 0°, 0°, 0°, 5°, 12°, 24°, in left support phase. Then, the twp 4-3-4 polynomials and forward kinematics can be used to draw out the trajectories of both right and left soles as shown in Fig. 10.

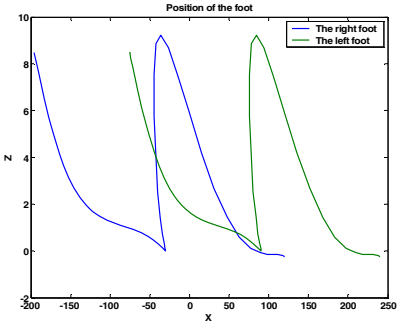


Fig. 10. The trajectories of both right and left soles as manipulating the biped robot in four DoFs

The 6-DoFs manipulation like the style of human walking is most easy planning in biped robot walking. Suppose a stride of the 6-DoFs manipulation is 330 mm. Let the set points of concerned joints be θ_2 = 26.5°, 18°, 0°, 0°, 0°, -15°, -30°, θ_3 = 10°, 7°, 0°, 0°, 0°, 25°, 40°, θ_5 = -36.5°, -20°, 0°, 0°, 0°, -7°, -10°, θ_8 = 10°, 10°, 15°, 20°, 25°, 30°, 36.5°, θ_{10} = -40°, -40°, -35°, -30°, -25°, -15°, -10°, θ_{11} = 30°, 30°, 25°, 20°, 10°, -15°, -26.5° in right support phase, and θ_2 = -30°, -30°, -25°, -20°, -10°, 15°, 26.5°, θ_3 = 40°, 40°, 35°, 30°, 25°, 15°, 10°, θ_5 = -10°, -10°, -15°, -20°, -25°, -30°, -36.5°, θ_8 = 36.5°, 20°, 0°, 0°, 0°, 7°, 10°, θ_{10} = -10°, -7°, 0°, 0°, 0°, -30°, -40°, θ_{11} = -26.5°, -18°, 0°, 0°, 0°, 15°, 30° in left support phase. Then, the twp 4-3-4 polynomials and forward kinematics can be used to draw out the trajectories of both right and left soles as shown in Fig. 11.

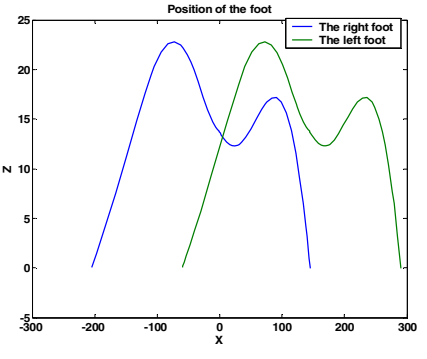


Fig. 11. The trajectories of both right and left soles as manipulating the biped robot in six DoFs

6 Analysis of the Motion Energy of the Biped Robot

The biped robot spends motion energy as maneuvering. It is well-known that the motion energy often involves kinetic and potential energy. The kinetic and potential energy, the spending of the biped robot, can be accumulated according to its parameters as shown in Table 1. The following first defines the kinetic and potential energy.

Kinetic energy is defined as

$$K = \frac{1}{2}mv^2 + \frac{1}{2}I\omega^2 \tag{12}$$

It is consisted of the kinetics of each link as follows

$$K_i = \frac{1}{2}m_i\left(\dot{x}_i^2 + \dot{z}_i^2\right) + \frac{1}{2}I_i\dot{\theta}_i^2 \tag{13}$$

Hence, the total kinetics is

$$K = \sum K_i = \sum \frac{1}{2}m_i\left(\dot{x}_i^2 + \dot{z}_i^2\right) + \sum \frac{1}{2}I_i\dot{\theta}_i^2 \tag{14}$$

where the inertia matrix is $I = \begin{bmatrix} I_{xx} & -I_{xy} & -I_{xz} \\ -I_{xy} & I_{yy} & -I_{yz} \\ -I_{xz} & -I_{yz} & I_{zz} \end{bmatrix}$, $I_{xx} = \frac{m}{3}\left(l^2 + h^2\right)$, $I_{yy} = \frac{m}{3}\left(w^2 + h^2\right)$, $I_{zz} = \frac{m}{3}\left(l^2 + w^2\right)$, $I_{xy} = \frac{m}{4}wl$, $I_{xz} = \frac{m}{4}hw$, $I_{yz} = \frac{m}{4}hl$

In addition, the potential energy is defined as

$$P = mgh \tag{15}$$

It is composed by the potential energy of each link as follows

$$P_i = m_i z_i g \tag{16}$$

Hence, the total potential energy is

$$P = \sum P_i = \sum m_i z_i g \tag{17}$$

The force equation is

$$F = mg + ma + I\alpha \tag{18}$$

The force of each link and joint is

$$F_i = m_i g + m_i\left(\ddot{x}_i + \ddot{z}_i\right) + I_i\ddot{\theta}_i \tag{19}$$

Hence, the total force is

$$F = \sum F_i = \sum m_i g + \sum m_i\left(\ddot{x}_i + \ddot{z}_i\right) + \sum I_i\ddot{\theta}_i \tag{20}$$

Suppose the swing leg moves from (x, z) = (-190mm, 58mm) to (120mm, 0mm). A stride of the biped robot is thus 310 mm. The manipulation of three different DoFs results in different trajectories of the swing leg's soles as shown in Fig. 12. In addition, the waist trajectories is drawn as shown in Fig. 13. The motion energy of the biped robot manipulated by the three DoFs is calculated according to these trajectories.

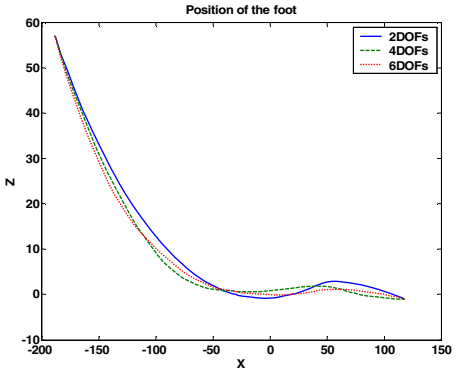


Fig. 12. The trajectories of soles as the biped robot manipulated by different DoFs

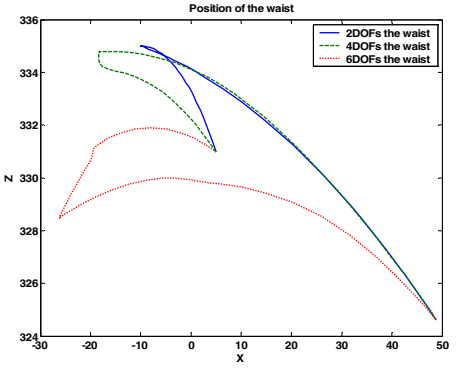


Fig. 13. The trajectories of waists as the biped robot manipulated by different DoFs

As shown in Fig. 12, the 6-DoFs manipulation achieves the smooth trajectory of the swing leg, but the longest trajectory of the waist as shown in Fig. 13. However, in contrast with the waist trajectories of two and four DoFs, that of six DoFs stays lower level. The lower waist trajectory often performs the larger regions of walking stability.

Fig. 14 compares the angle, angular velocity and angular acceleration of the waist joint as the manipulation of different DoFs. As shown in Fig. 14, the three trajectories is almost consistent, but the trajectory of 6-DoFs manipulation is following the traces between those of 2-DoFs and 4-DoFs manipulation. i. e. The trajectory of 6-DoFs is not too slow and not too fast.

Finally, the motion energy of the biped robot manipulated by the three DoFs is compared. First, the kinetic energy is compared in Fig. 15. Fig. 15 shows the trajectories of the kinetics and force changed with respect to time. As shown in Fig. 15, it is amazing that the 2-DoFs manipulation almost spends the least energy except the swing leg in the period on ground. In the comparison of total spending kinetics, the manipulation of two, four and six DoFs are 0.0713 (J), 0.0735 (J), and 0.0771 (J),

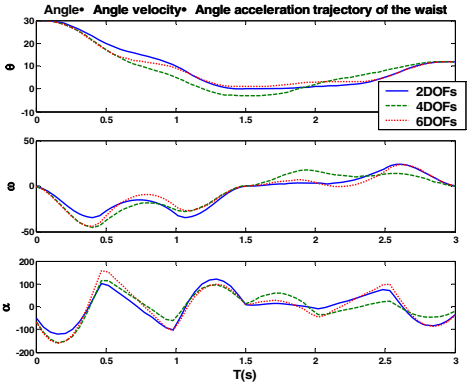


Fig. 14. The angle, angular velocity and angular acceleration of the waist of different DoFs

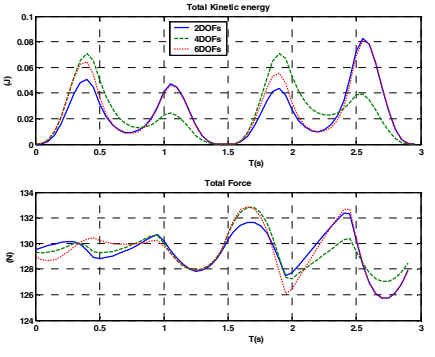


Fig. 15. The kinetics and force trajectories of different DoFs

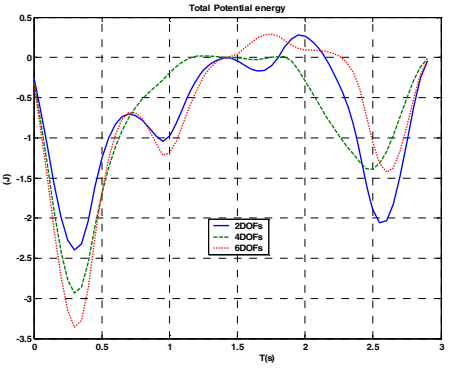


Fig. 16. The potential trajectories of different DoFs

respectively. It is obvious that two DoFs is of the least spending. Then the potential energy is compared in Fig. 16. Fig. 16 shows the trajectories of potential energy changed with respect to time. As shown in Fig. 16, there is no obvious difference of the three different DoFs, but as comparing with the change range of potential energy, the 6-DoFs manipulation is the largest one. As comparing the different potential between initial and final locations, the manipulation of two, four and six DoFs is -2.1923 (J), -2.1927 (J) and -2.1921 (J), respectively. They are almost consistent, because their initial and final locations are the same. In summary, the 2-DoFs manipulation spends the least kinetic energy because of the simplest structure. But the simplest structure makes unsmooth motion. Four and six DoFs let the manipulation be smooth, but spend much kinetic energy, and make the operation more complicated.

7 Conclusion and Further Development

It is interested to study the difference of a biped robot manipulated by different DoFs. In this study the analysis and comparison aim at the point of view form motion energy. It is amazing for the study results of the biped robot manipulated by two, four and six DoFs, respectively. Even if the 6-DoFs manipulation performs smooth trajectory, the spending of kinetic energy is the largest one. The 2-DoFs manipulation achieves the constrained walking, the spending of motion energy is the smallest one. In addition to the smallest motion energy, 2-DoFs biped robots possess the advantages of simple structure and easy control. It is great hopeful for the development of 2-DoFs biped robots in the future.

Biped robots are the integration research including mechanism design, motor control, sensing fusion, and so on. It is worthful to study from the infrastructure to advance of the biped robot development.

References

1. Harada, K., Kajita, S., Kaneko, K., Hirukawa, H.: An Analytical Method on Real-time Gait Planning for a Humanoid Robot. International Journal of Humanoid Robotics (October 8, 2004)
2. Asano, F., Luo, Z.-W., Yamakita, M.: Biped Gait Generation and Control Based on a Unified Property of Passive Dynamic Walking. IEEE Transactions on Robotics 21(4), 754–762 (2005)
3. Agrawal, S.K., Fattah, A.: Motion Control of a Novel Planar Biped With Nearly Linear Dynamics. IEEE/ASME Transactions on Mechatronics 11(2), 162–168 (2006)
4. Shih, C.-L.: Ascending and Descending Stairs for a Biped Robot. IEEE Transactions on Systems, Man, and Cybernetics — Part A: Systems and Humans 29(3), 255–268 (1999)
5. Huang, Q., Yokoi, K., Kajita, S., Kaneko, K., Arai, H., Koyachi, N., Tanie, K.: Planning Walking Patterns for a Biped Robot. IEEE Transactions on Robotics and Automation 17(3), 280–289 (2001)
6. Huang, Q., Nakamura, Y.: Sensory Reflex Control for Humanoid Walking. IEEE Transactions on Robotics 21(5), 977–984 (2005)

Footstep Planning Based on Univector Field Method for Humanoid Robot

Youngdae Hong and Jong-Hwan Kim

Department of Electrical Engineering and Computer Science, KAIST
Daejeon, Korea
{ydhong,johkim}@rit.kaist.ac.kr
http://rit.kaist.ac.kr

Abstract. This paper proposes a footstep planning algorithm based on univector field method optimized by evolutionary programming for humanoid robot to arrive at a target point in a dynamic environment. The univector field method is employed to determine the moving direction of the humanoid robot at every footstep. Modifiable walking pattern generator, extending the conventional 3D-LIPM method by allowing the ZMP variation while in single support phase, is utilized to generate every joint trajectory of a robot satisfying the planned footstep. The proposed algorithm enables the humanoid robot not only to avoid either static or moving obstacles but also step over static obstacles. The performance of the proposed algorithm is demonstrated by computer simulations using a modeled small-sized humanoid robot HanSaRam (HSR)-VIII.

Keywords: Footstep planning, univector field method, evolutionary programming, humanoid robot, modifiable walking pattern generator.

1 Introduction

These days research on a humanoid robot has made rapid progress for dexterous motions with the hardware development. Various humanoid robots have demonstrated stable walking with control schemes [1]-[5]. Considering the future of the humanoid robot as a service robot, research on navigation in indoor environments such as homes and offices with obstacles is now needed.

In indoor environments, most of research on navigation has been carried out for differential drive mobile robots. The navigation method of the mobile robots is categorized into separated navigation and unified navigation. The separated navigation method, such as structural navigation and deliberative navigation, separates path planning and path following as two isolated tasks. In the path planning step, a path generation algorithm is developed which connects the staring point with the end point without crossing the obstacles. To find the shortest path many searching algorithms such as A* algorithm and dynamic programming have been applied [6]. On the other hand, in unified navigation method such as the artificial potential field method [7], [8], the path planning step and the path following step are unified in one task.

In the navigation research, differential drive mobile robots make a detour to avoid obstacles to arrive at a goal position. On the other hand, humanoid robots are able to

J.-H. Kim et al. (Eds.): FIRA 2009, LNCS 5744, pp. 125–134, 2009.

traverse obstacles by their legs. When they move around in an environment, positions of their footprints are important as there are obstacles. Thus, the study of footstep planning for humanoid robots is an important research issue.

As research on footstep planning, algorithm obtaining information of obstacle's shape and location by sensors was presented [9]. Through obtained information, a robot determines its step length which is predefined as three type step lengths and its motion such as circumventing, stepping over or stepping on obstacles. Also, an algorithm finding alternative path employing A* by heuristic cost function was developed [10]. Stable region of robot's footprints is predetermined and then a few of placements of them are selected as a discrete set. This algorithm checks collision between a robot and obstacles by 2D polygon intersection test. Human-like strategy for footstep planning was also presented [11].

In this paper, a footstep planning algorithm based on the univector field method for humanoid robot is proposed. The univector field method is one of the unified navigation methods, which is designed for fast differential drive mobile robots to enhance performances. Using this method, robot can navigate rapidly to the desired position and orientation without oscillations and unwanted inefficient motions [12], [13]. The footstep planning algorithm determines moving direction of a humanoid robot in real time and has low computing cost by employing the univector field method. Besides, it is able to modify foot placement depending on obstacle's position. Inputting the moving direction and step length of a robot at every footstep to modifiable walking pattern generator [14], every joint trajectory is generated. The proposed algorithm generates an evolutionary optimized path by evolutionary programming (EP) considering hardware limit of a robot and makes a robot arrive at a goal with desired direction. Computer simulations are carried out by a model of HanSaRam (HSR)-VIII which is a small-sized humanoid robot developed in Robot Intelligence Technology (RIT) Lab, KAIST.

The rest of the paper is organized as follows: Section 2 describes an overview of univector field method and Section 3 explains MWPG. In Section 4 a footstep planning algorithm is proposed. Computer simulation results are presented in Section 5. Finally concluding remarks follow in Section 6.

2 Univector Field Method

The univector field method is one of path planning methods developed for a differential drive mobile robot. The univector field consists of *move-to-goal univector field* which leads a robot to move to a destination and *avoid-obstacle univector field* which makes a robot avoid obstacles. Its moving direction is decided by combining move-to-goal univector field and avoid-obstacle univector field. The univector field method requires relatively low computing power because it does not generate a whole path from a start point to a destination before moving, but generates a moving direction decided at every step in real time. In addition, it is easy to plan a path in a dynamic environment with moving obstacles. Thus, this method of path planning is adopted and extended for a humanoid robot.

2.1 Move-to-Goal Univector Field

The move-to-goal univector field is defined as

$$\mathbf{v}_{muf} = [-\cos(\theta_{muf}) \ -\sin(\theta_{muf})]^T, \tag{1}$$

where

$$\theta_{muf} = \cos^{-1}(\frac{p_x - g_x}{d_{goal}}), d_{goal} = \sqrt{(p_x - g_x)^2 + (p_y - g_y)^2},$$

θ_{muf} is the angle from x-axis of the goal at robot's position, d_{goal} is the distance between the center of a goal and robot's position, and (p_x, p_y) and (g_x, g_y) are the robot's position and the goal position, respectively.

2.2 Avoid-Obstacle Univector Field

The avoid-obstacle univector field is defined as

$$\mathbf{v}_{auf} = [\cos(\theta_{auf}) \ \sin(\theta_{auf})]^T, \tag{2}$$

where

$$\theta_{auf} = \cos^{-1}(\frac{p_x - o_x}{d_{ob}}), d_{ob} = \sqrt{(p_x - o_x)^2 + (p_y - o_y)^2},$$

θ_{auf} is the angle from x-axis of an obstacle at robot's position, d_{ob} is the distance between the center of an obstacle and robot's position and (o_x, o_y) is the position of an obstacle.

Total univector field is determined by properly combining the move-to-goal univector field and the avoid-obstacle univector field. Total univector $\mathbf{v}_{tuf}$ is defined as

$$\mathbf{v}_{tuf} = w_{muf}\mathbf{v}_{muf} + w_{auf}\mathbf{v}_{auf}, \tag{3}$$

where w_{muf} and w_{auf} represent the scale factor of the move-to-goal univector field and the avoid-obstacle univector field, respectively.

3 Modifiable Walking Pattern Generator

The modifiable walking pattern generator (MWPG) extended the conventional 3D-LIPM method by allowing the ZMP variation while in single support phase. In the conventional 3D-LIPM without the ZMP variation, only the homogeneous solutions of the 3D-LIPM dynamic equation were considered. However, considering the particular solutions, more extensive and unrestricted walking patterns could be generated by allowing the ZMP variation. The solutions with both homogeneous and particular parts are as follows:
Sagittal motion:

$$\begin{bmatrix} x_f \\ v_f T_c \end{bmatrix} = \begin{bmatrix} C_T & S_T \\ S_T & C_T \end{bmatrix} \begin{bmatrix} x_i \\ v_i T_c \end{bmatrix} - \frac{1}{T_c} \begin{bmatrix} \int_0^T S_t \bar{p}(t) dt \\ \int_0^T C_t \bar{p}(t) dt \end{bmatrix}, \tag{4}$$

Lateral motion:

$$\begin{bmatrix} y_f \\ w_f T_c \end{bmatrix} = \begin{bmatrix} C_T & S_T \\ S_T & C_T \end{bmatrix} \begin{bmatrix} y_i \\ w_i T_c \end{bmatrix} - \frac{1}{T_c} \begin{bmatrix} \int_0^T S_t \bar{p}(t) dt \\ \int_0^T C_t \bar{p}(t) dt \end{bmatrix}, \tag{5}$$

where $(x_i, v_i)/(x_f, v_f)$ and $(y_i, w_i)/(y_f, w_f)$ represent initial/final position and velocity of the CM in the sagittal and lateral plane, respectively. S_t and C_t are defined as $\cosh(t/T_c)$ and $\sinh(t/T_c)$ with time constant $T_c = \sqrt{Z_c/g}$. The functions $p(t)$ and $q(t)$ are ZMP trajectories for the sagittal and lateral planes, respectively. $\bar{p}(t) = p(T-t)$ and $\bar{q}(t) = q(T-t)$. Through the variation of the ZMP, the walking state (WS), which is the state of the point mass in the 3D-LIPM represented in terms of CM position and linear velocity can be moved to the desired WS in the region of possible trajectories expanded by applying the particular solutions. By means of the MWPG, a humanoid robot can change both sagittal and lateral step lengths, rotation angle of ankles and the period of the walking pattern [14].

4 Footstep Planning Algorithm

In this section, a footstep planning algorithm for a humanoid robot is described. It decides moving orientation at every footstep by univector field navigation method. Using the determined orientations, it calculates exact foot placement. Subsequently, by inputting the moving direction and step length of a robot at every footstep by proposed footstep planning algorithm to MWPG, every joint trajectory is generated to satisfy the planned footstep.

4.1 Path Planning

To apply univector field method to the path generation of a humanoid robot, the following three issues are considered. To generate a natural and effective path, obstacle's boundary and virtual obstacle [15] are introduced to the avoid-obstacle univector field considering the obstacle's size and movement, respectively. Also, a hyperbolic spiral univector field is developed as a move-to-goal univector field in order to reach a destination with a desired orientation [13].

Boundary of Avoid-Obstacle Univector Field. The repulsive univector field by obstacles is not generated at every position but generated in a restricted range by applying a boundary to the avoid-obstacle univector field. Also, the more the robot's position becomes distant from the center of an obstacle, the more the magnitude of the repulsive univector field decreases linearly. Consequently, a robot is not influenced the repulsive univector field at the region which is away from the boundary of obstacles. Considering this boundary effect, the avoid-obstacle univector $\mathbf{v}_{auf}$ is defined as

$$\mathbf{v}_{auf} = k_b[\cos(\theta_{auf}) \ \sin(\theta_{auf})]^T \tag{6}$$

where

$$k_b = \frac{d_{boun} - (d_{ob} - o_{size})}{d_{boun}},$$

o_{size} is the obstacle's radius, d_{boun} is the size of boundary and k_b is a scale factor. By introducing the boundary into the avoid-obstacle univector field, an effective path is generated.

Virtual Obstacle. The virtual obstacle is defined by introducing a shifting vector to the center position of a real obstacle, where the direction of shifting vector is opposed to the robots moving direction and the magnitude is proportional to the robots moving velocity. Then, the position of the center of the virtual obstacle is obtained as

$$[o_x^{virtual}\ o_y^{virtual}]^T = [o_x^{real}\ o_y^{real}]^T + \mathbf{s}, \tag{7}$$

$$\mathbf{s} = -k_v \mathbf{v}_{robot},$$

where $(o_x^{virtual}, o_y^{virtual})$ is the virtual obstacle's position, (o_x^{real}, o_y^{real}) is the real obstacle's position, $\mathbf{s}$ is the shifting vector, k_v is the scale factor of the virtual obstacle and $\mathbf{v}_{robot}$ is the robot's velocity vector. When calculating the avoid-obstacle univector, the virtual obstacle's positions are used instead of the real obstacles. By introducing the virtual obstacle, a robot can avoid obstacles more safely and smoothly by a generated path at every step.

Hyperbolic Spiral Univector Field. The move-to-goal univector field is designed by the hyperbolic spiral for a robot to get to a target point with a desired orientation. The hyperbolic spiral univector field $\mathbf{v}_{huf}$ is defined as

$$\mathbf{v}_{huf} = [\cos(\phi_h)\ \sin(\phi_h)]^T, \tag{8}$$

where

$$\phi_h = \begin{cases} \theta \pm \frac{\pi}{2}(2 - \frac{d_e + k_r}{\rho + k_r}) & \text{if } \rho > d_e \\ \theta \pm \frac{\pi}{2}\sqrt{\frac{\rho}{d_e}} & \text{if } 0 \le \rho \le d_e \end{cases},$$

θ is the angle from x-axis of the goal at robot's position. The notation $\pm$ represents the direction of movement, where + is when a robot moves clockwise and - counter clockwise. k_r is an adjustable parameter. If k_r becomes larger, the maximal value of curvature derivative decreases and the contour of the spiral becomes smoother. ρ is the distance between the center of the destination and robot's position d_e is predefined radius that decides the size of the spiral.

By designing a move-to-goal univector field with hyperbolic spiral, a robot can arrive at a destination with any orientation angle. In this paper, in order to obtain the desired posture at a target position, two hyperbolic spiral univector fields are combined. The move-to-goal univector field is defined as

$$\phi_{muf} = \begin{cases} \theta_{up} + \frac{\pi}{2}(2 - \frac{d_e + k_r}{\rho_{up} + k_r}) & \text{if } p_y^h > g_{size} \\ \theta_{down} - \frac{\pi}{2}(2 - \frac{d_e + k_r}{\rho_{down} + k_r}) & \text{if } p_y^h < -g_{size} \\ \theta_{dir} & \text{otherwise} \end{cases}, \tag{9}$$

with

$$\rho_{up} = \sqrt{p_x^{h2} + (p_y^h - d_e - g_{size})^2},\ \rho_{down} = \sqrt{p_x^{h2} + (p_y^h + d_e + g_{size})^2},$$

$$\theta_{up} = \tan^{-1}\left(\frac{p_y^h - d_e - g_{size}}{p_x^h}\right) + \theta_{dir},\ \theta_{down} = \tan^{-1}\left(\frac{p_y^h + d_e + g_{size}}{p_x^h}\right) + \theta_{dir},$$

$$\mathbf{p}^h = \mathbf{M}_{rot}\mathbf{M}_{trans}\mathbf{p},$$

$$\mathbf{M}_{trans} = \begin{bmatrix} 1 & 0 & -g_x \\ 0 & 1 & -g_y \\ 0 & 0 & 1 \end{bmatrix},\ \mathbf{M}_{rot} = \begin{bmatrix} \cos(-\theta_{dir}) & -\sin(-\theta_{dir}) & 0 \\ \sin(-\theta_{dir}) & \cos(-\theta_{dir}) & 0 \\ 0 & 0 & 1 \end{bmatrix},$$

$$\mathbf{p} = \begin{bmatrix} p_x & p_y & 1 \end{bmatrix}^T,\ \mathbf{p}^h = \begin{bmatrix} p_x^h & p_y^h & 1 \end{bmatrix}^T,$$

where g_{size} is the radius of the goal region and θ_{dir} is the desired arrival angle at a target. By using the move-to-goal univector field which is composed of the hyperbolic spiral univector fields, a robot can arrives at a goal with any arrival angles.

4.2 Footstep Planning

While a humanoid robot moves towards a destination, there is a situation when it has to step over an obstacle if it is not too high. This is the main difference from the path planning for a differential drive mobile robot, as it tries to find a detour route to circumvent obstacles instead of stepping over them. In this section, a footstep planning algorithm is proposed, which enables a robot to traverse over the obstacles effectively.

It is very natural and efficient way that a robot steps over them instead of detouring, if its moving direction is maintained. The proposed algorithm enables a robot step over the obstacles with minimal step length while maintaining its moving direction. It is

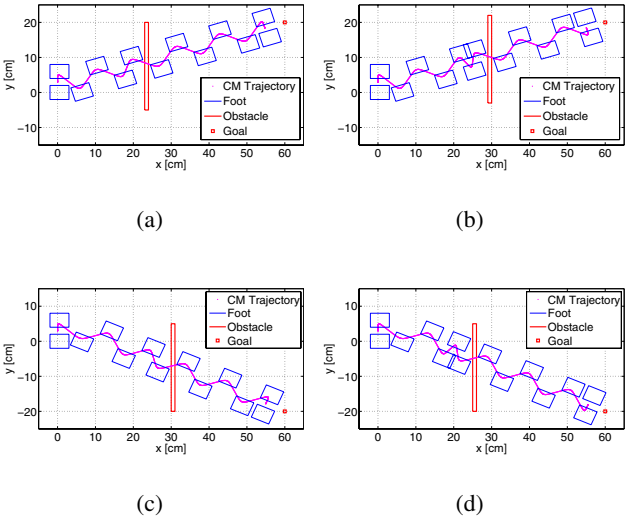


Fig. 1. Stepping over an obstacle. (a) Left leg is supporting leg without additional step (b) Left leg is supporting leg with additional step (c) Right leg is supporting leg without additional step (d) Right leg is supporting leg with additional step.

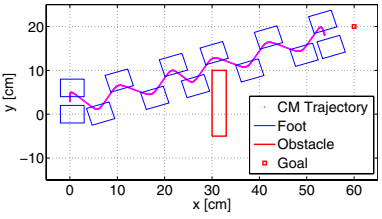


Fig. 2. Stepping over an obstacle when an obstacle is in front of one leg

assumed that the shape of obstacles is a rectangle with narrow width and long length as shown in Fig. 1.

The forward and backward step length from a supporting leg of a humanoid robot are restricted because of hardware limitation. If an obstacle is wider in width than the maximum step length of a humanoid robot, it is not able to step over an obstacle. Thus, a humanoid robot has to step over an obstacle with the shortest possible step length in order to step over the widest possible obstacle. The step length of a humanoid robot is determined by which leg is a supporting leg when it steps over an obstacle. As the proposed algorithm considers these facts, it enables a robot to step over obstacles with the shortest step length. Fig. 1 shows the footprints to step over an obstacle using this algorithm. Fig. 1(a) and Fig. 1(d) are situations when a left foot comes close to the obstacle earlier than a right foot and Fig. 1(b) and Fig. 1(c) are situations when a right foot approaches the obstacle closely than the other one. In case of Fig. 1(a) and 1(b), the left leg is appropriate as a supporting leg for the minimum step length. On the other hand, the right leg is appropriate as a supporting leg in Fig. 1(c) and 1(d). Therefore, in order to make a left leg as a supporting leg in Fig. 1(b) and a right leg as a supporting leg in Fig. 1(d), one more step is needed before stepping over the obstacle, while such an additional step is not needed in Fig. 1(a) and 1(c).

There is a situation when an obstacle is only in front of one leg such that the other leg can be placed without considering the obstacle. The proposed algorithm deals with this situation such that it can step over the obstacle effectively like a human being. Fig. 2 shows the footprints of a robot in this case.

4.3 Parameter Optimization by Evolution Programming

A humanoid robot has the constraint of change in rotation of legs on account of the hardware limitation. Hence, when planning footsteps for a biped robot by the proposed algorithm, the maximum change in rotation of legs has to be assigned. In this algorithm, there are seven parameters to be assigned such as k_v in the virtual obstacle, d_{boun} in the avoid-obstacle univector field, d_e, k_r, g_{size} in the move-to-goal univector field and w_{muf}, w_{auf} in composition of the move-to-goal univector field and the avoid-obstacle univector field, respectively. A robot can arrive at a goal with the change in rotation of legs within constraints by selecting appropriate values of parameters mentioned above. Also to generate the most effective path, EP is employed to choose the values of parameters. The fitness function in EP is designed considering the followings:

- A robot should arrive at a destination with a minimum position error.
- The facing direction of a robot at a destination should be the desired one.
- A robot should not collide with obstacles.
- The change in rotation of legs should not exceed the constraint value.

Consequently, the fitness function is defined as

$$f = -(k_p P_{err} + k_q \mid \theta_{err} \mid + k_{col} N_{col} + k_{const} N_{const}) \tag{10}$$

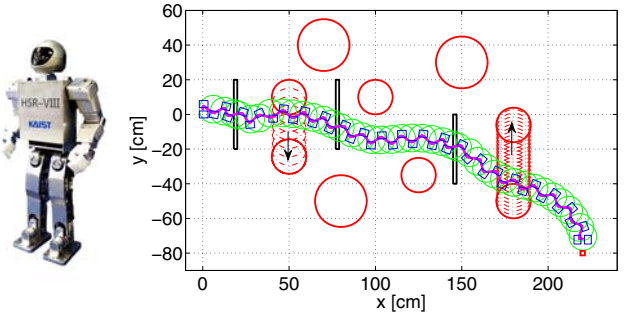
where N_{const} is the number of constraint violations of change in rotation of legs, N_{col} is the number of obstacle collisions of the robot, θ_{err} is the difference between the desired orientation and the orientation of a robot at a goal, P_{err} is the position error at a goal and $k_{const}, k_{col}, k_q, k_p$ are constants.

5 Simulation Results

HSR-VIII (Fig. 3(a)) is a small-sized humanoid robot that has been continuously undergoing redesign and development in RIT Lab, KAIST since 2,000. Its height and weight are 52.8 cm and 5.5 kg, respectively. It has 26 DOFs which consists of 12 DC motors with harmonic drives for reduction gears in the lower body and 14 RC servo motors in the upper body. HSR-VIII was modeled by Webot which is the 3D mobile robotics simulation software [16]. Simulations were carried out with Webot of the HSR-VIII model by applying the proposed footstep planning algorithm.

Through the simulation, seven parameters in the algorithm were optimized by EP. Maximum rotating angle of the robot's ankles was selected heuristically as 40°. After 100 generations, the parameters were optimized as k_v=1.94, d_{boun}=20.09, d_e=30.04, k_r=0.99, g_{size}=0.94, w_{muf}=1.96, w_{auf}=1.46.

Fig. 3(b) shows the sequence of robot's footsteps as a 2D simulation result, where there were ten obstacles of three different kinds such as five static circular obstacles



(a) (b)

Fig. 3. (a) HSR-VIII. (b) Sequence of footsteps in the environment with ten obstacles of three different kinds.

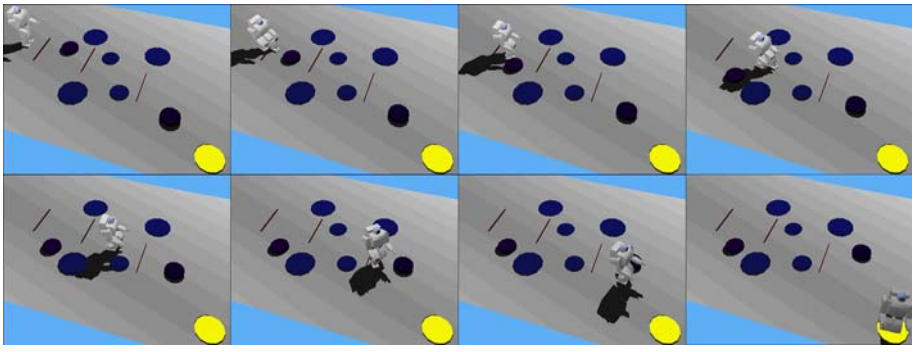


Fig. 4. Snap shots of 3D simulation result by Webot in the environment with ten obstacles of three different kinds. (A goal is a circle in the right bottom corner.)

and two moving circular obstacles and three static rectangular obstacles with a height of 1.0 cm. The desired angle at a destination was fixed at 90° from x-axis. As shown in the figure, by the proposed algorithm the robot moves from a start point to a target goal in the right bottom corner, while avoiding static and moving circular obstacles and stepping over static rectangular ones by adjusting its step length. In addition, the robot faces the desired orientation at the goal. Fig. 5 shows the 3D simulation result by Webot, where the environment is the same as that used in the 2D simulation. Similar result was obtained as in Fig. 3(b). In particular, in third and sixth snapshots of the Fig. 10, it can be seen that the robot makes a turn before colliding with the moving circular obstacles predicting their movement.

6 Conclusion

The real-time footstep planning algorithm was proposed for a humanoid robot to travel to a destination avoiding and stepping over obstacles. The univector field method was adopted to determine the heading direction and using the determined orientations, exact foot placement was calculated. The proposed algorithm generated the efficient path by applying a boundary to the avoid-obstacle univector field and introducing the virtual obstacle concept. Furthermore, it enables a robot to get to a destination with a desired orientation by employing the hyperbolic spiral univector field. The proposed algorithm made a robot possible to step over an obstacle with minimal step length maintaining its heading orientation. It also considered the situation when an obstacle is in front of only one leg. In this case, it steps over the obstacle while placing the other leg properly as a supporting one. The effectiveness of the algorithm was demonstrated by computer simulations in dynamic environment. As a further work, experiments with a real small-sized humanoid robot HSR-VIII will be carried out using a global camera to demonstrate the applicability of the proposed algorithm.

References

1. Nishiwaki, K., Sugihara, T., Kagami, S., Kanehiro, F., Inaba, M., Inoue, H.: Design and Development of Research Platform for Perception- Action Integration in Humanoid Robot: H6. In: Proc. IEEE/RSJ Int. Conference on Intelligent Robots and Systems, pp. 1559–1564 (2000)
2. Kaneko, K., Kanehiro, F., Kajita, S., Hirukawa, H., Kawasaki, T., Hirata, M., Akachi, K., Isozumi, T.: Humanoid Robot HRP-2. In: Proc. IEEE Int'l. Conf. on Robotics and Automation, ICRA 2004 (2004)
3. Sakagami, Y., Watanabe, R., Aoyama, C., Matsunaga, S., Higaki, N., Fujimura, K.: The intelligent ASIMO: system overview and integration. In: Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, pp. 2478–2483 (2002)
4. Ogura, Y., Aikawa, H., Shimomura, K., Kondo, H., Morishima, A., Lim, H., Takanishi, A.: Development of a New Humanoid Robot WABIAN-2. In: Proc. IEEE Int'l. Conf. on Robotics and Automation, ICRA 2006 (2006)
5. Kim, Y.-D., Lee, B.-J., Ryu, J.-H., Kim, J.-H.: Landing Force Control for Humanoid Robot by Time-Domain Passivity Approach. IEEE Trans. on Robotics 23(6), 1294–1301 (2007)
6. Kanal, L., Kumar, V. (eds.): Search in Artificial Intelligence. Springer, New York (1988)
7. Borenstein, J., Koren, Y.: Real-time obstacle avoidance for fast mobile robots. IEEE Trans. Syst., Man, Cybern. 20, 1179–1187 (1989)
8. Borenstein, J., Koren, Y.: The vector field histogram-fast obstacle avoidance for mobile robots. IEEE Trans. Syst., Man, Cybern. 7, 278–288 (1991)
9. Yagi, M., Lumelsky, V.: Biped Robot Locomotion in Scenes with Unknown Obstacles. In: Proc. IEEE Int'l. Conf. on Robotics and Automation (ICRA 1999), Detroit, MI, May 1999, pp. 375–380 (1999)
10. Chestnutt, J., Lau, M., Cheung, G., Kuffner, J., Hodgins, J., Kanade, T.: Footstep planning for the honda asimo humanoid. In: Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, pp. 631–636 (2005)
11. Ayaz, Y., Munawar, K., Bilal Malik, M., Konno, A., Uchiyama, M.: Human-Like Approach to Footstep Planning Among Obstacles for Humanoid Robots. In: Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, pp. 5490–5495 (2006)
12. Kim, Y.-J., Kim, J.-H., Kwon, D.-S.: Evolutionary Programming-Based Uni-vector Field Navigation Method for Fast Mobile Robots. IEEE Trans. on Systems, Man and Cybernetics - Part B - Cybernetics 31(3), 450–458 (2001)
13. Kim, Y.-J., Kim, J.-H., Kwon, D.-S.: Univector Field Navigation Method for Fast Mobile Robots. Korea Advanced Institute of Science and Technology, Ph.D. Thesis
14. Lee, B.-J., Stonier, D., Kim, Y.-D., Yoo, J.-K., Kim, J.-H.: Modifiable Walking Pattern of a Humanoid Robot by Using Allowable ZMP Variation. IEEE Transaction on Robotics 24(4), 917–925 (2008)
15. Lim, Y.-S., Choi, S.-H., Kim, J.-H., Kim, D.-H.: Evolutionary Univector Field-based Navigation with Collision Avoidance for Mobile Robot. In: Proc. 17th World Congress The International Federation of Automatic Control, Seoul, Korea (July 2008)
16. Michel, O.: Cyberbotics Ltd. WebotsTM: Professional mobile robot simulation. Int. J. of Advanced Robotic Systems 1(1), 39–42 (2004)

Tendon Based Full Size Biped Humanoid Robot Walking Platform Design

Chung-Hsien Kuo and Kuo-Wei Chiou

Department of Electrical Engineering,
National Taiwan University of Science and Technology,
106 Taipei, Taiwan, R.O.C.
{chkuo,dream10013}@mail.ntust.edu.tw

Abstract. Actuators and gear trains of most biped humanoid robots are divergently allocated on the links of two legs. Disadvantages of such a mechanical design are complicated wiring of power cord and sensing/ control signal bundles and imprecise kinetics models of mixed link-and-actuator structures. Based on these drawbacks, this paper proposes a tendon-driven mechanism to develop a lower body structure of a full-size biped humanoid robot. The actuators are compacted as an actuator module, and they are placed at a distal site. A 12 degree-of-freedom mechanical structure is proposed with 100 cm in height and 45 kg in weight. The gait planning module is simulated and evaluated using the Matlab software. At the same time, an ARM7 based controller is developed to automatically generate walking patterns as well as to control the motors. Finally, a tendon-driven biped humanoid robot prototype is realized for practical waling control in the future.

Keywords: tendon driven mechanism; gait planning; biped humanoid robots; embedded controller.

1 Introduction

Biped humanoid robots are challenging research topics, especially for full size humanoid robots. From the viewpoints of motion behaviors, biped humanoid robots have similar motion patterns with humans; therefore, they are capable to walk on uneven grounds. However, due to complicated mechanical structures and large number of actuators, the control of biped robots is very difficult. The reasons are body links and actuators of most of famous full size humanoid robots such as, Wabian [1], Asimo [2] and HRP-4c [3], are coupling constructed, and the drawback of such coupled designs are:

1. Coupled designs of body links and actuators results in impact on actuators directly when the foot touches the ground. The vibrations of hard contact may result in damaging the actuators as well as inducing instability of control systems.
2. Coupled designs of body links and actuators are more difficult to model the center of gravity (COG) and zero point (ZMP) of robots. Such a situation decreases the precision of dynamics modeling as well as increases the efforts of developing walking control systems.

J.-H. Kim et al. (Eds.): FIRA 2009, LNCS 5744, pp. 135–150, 2009.

3. Coupled designs of body links and actuators have to deal with complicated wiring efforts on signals, controls and powers for a centralized control system. A huge bundle of wires is not so feasible for creating a humanoid robot.

On the other hand, a tendon based actuation system is desired to drive a distal joint using a set of tendon wires. Due to non-rigid connection between joints and actuators, the configuration of mechanical structure represents more flexible solutions. Therefore, this paper develops a full size humanoid robot using the tendon based actuation system based on the following features:

1. Non-rigid connections between body and actuator reduce the impacts and vibrations when the foot touches the ground.
2. Flexible configurations of distal joints may pack all actuators in a module unit so that the wiring efforts for distal actuators can be reduced. At the same time, the weight distributions of body links would be much simpler than coupled ones. The packed actuator module unit may be regards as a unit mass to increase the COG and ZMP calculations as well as to simplify the dynamics models of humanoid robots.
3. Distal joint actuation systems may perform more nice-looking robot bodies.

In order to verify the proposed tendon based full size biped humanoid robot walking platform, a 12 degree-of-freedom mechanical structure is designed using the Pro/Engineer 3D computer aided design (CAD) software. The biped walking platform is composed of lower body linkages of two legs and an actuator module unit containing 12 motor actuators. To eliminate backlashes of motions, 6 harmonic drive gear motors are used. In addition, 10 distal joints are actuated using tendon wires. Two actuators are directed coupled at the hip joints. These two actuators also serve as the interfaces of two legs and the actuator module unit.

The height of the robot structure is 100 cm in height and 45 kg in weight, where the actuator module unit is 30 kg. Based on the tendon based actuation design, two legs are constructed as 15 kg, and it can be more close to an ideal light weight leg inverted pendulum system. In this manner, the constructions of walking control system would be much simple.

On the other hand, the gait pattern planning [4] module is also constructed to simulate the waling behavior using the Matlab. The simulated gait pattern data are delivered to an ARM (advanced RSIC machine) 7 based gait motion controller. The based gait motion controller is responsible of retrieving gait patterns as well as controlling positions of motor actuators. Although the proposed tendon based actuation design is a part of an ongoing project; however, a preliminary control experiment of the robot is also briefly discussed.

Finally, this paper is organized as follows: section 2 introduces the mechanical design of the tendon based full size biped humanoid robot walking platform; section 3 illustrates the kinematics of the proposed walking platform; section 4 describes the gait pattern generation; section 5 presents the implementations and experiments; and finally, the conclusions and future works are concluded in section 6.

2 Mechanical Design

This paper proposed a tendon based full size biped humanoid robot walking platform. This platform is developed based decoupling the actuators and body links to perform previously mentioned features. The mechanical structure of the robot is designed using the Pro/E commercial software. The Pro/E is a visualized 3D CAD tool. In this section, the robot body structure and their realizations are illustrated in the following subsections.

2.1 Robot Body Structure Design

In this paper, the tendon based full size biped humanoid robot is constructed as 12 degree of freedoms. There are 3 degree of freedoms desired for each hip joint; 1 degree of freedom desired for each knee joint; and 2 degree of freedoms desired for each ankle joint, as shown in Fig. 1. The hip and ankle joints are designed as diagonal configurations so that the motion behaviors would be similar with humans' motions.

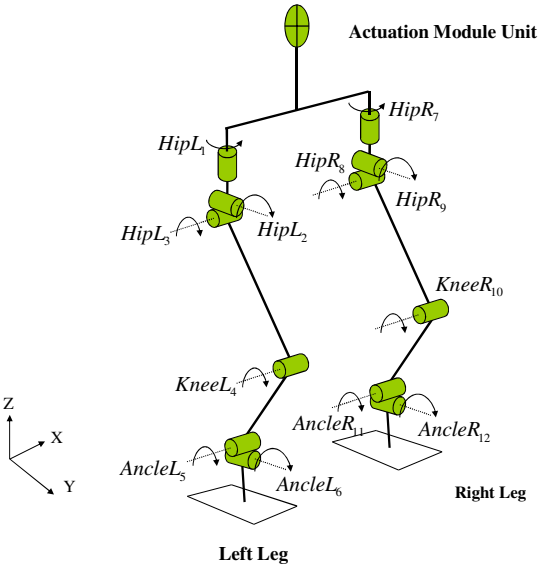


Fig. 1. Mechanical structure of the proposed biped walking platform

In this mechanical structure, 10 degree of freedoms of $L2$ to $L6$ and $R8$ to $R12$ are distally actuated; and 2 degree of freedoms of $L1$ and $R7$ are directly actuated. It is obvious that $L1$ and $R7$ serve as the interfaces of two legs and the actuator module unit. Note that the actuator module unit is modeled as a unit mass. In addition to the mechanical structure, the dimensions (in mm) of the robot body of front and lateral views are also indicated in Fig. 2. Based on the dimension specification, the proposed biped robot is 100 cm in height.

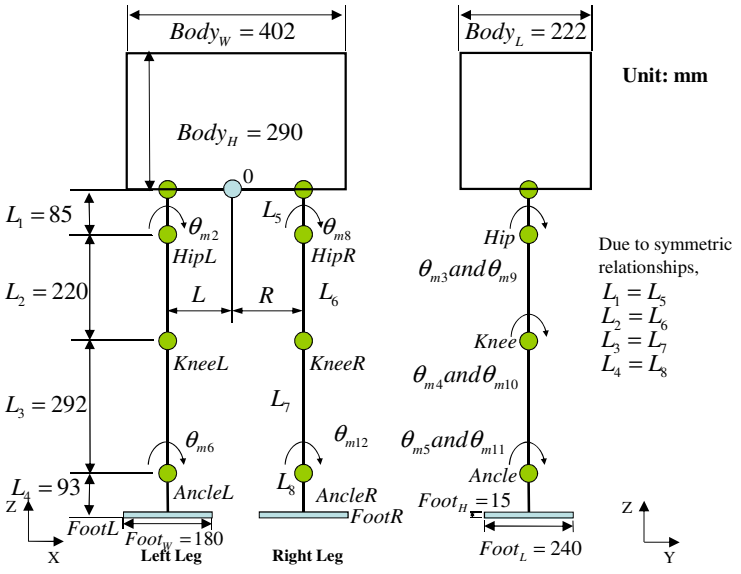


Fig. 2. Front and lateral views of the robot body dimensions

2.2 Realization of Robot Body

Based on the structure and dimension specifications, the detailed CAD models of two legs and an actuation module unit are designed. Due to decoupling design of actuators and distal joints, the CAD models of two legs are quite simple except the design of tendon pulleys. Detailed specifications of tendon slot width and depth, diameter of tendon pulley, number encountered must be carefully dealt with to perform reliable operations. The realizations of robot body can be categorized as "above knee (AK) leg mechanism", "below knee (BK) leg mechanism" and "actuation module unit".

The CAD of the AK leg mechanism is shown in Fig. 3. In this figure, a supporting cubic is designed as the frames of two diagonal tendon pulleys to form the hip joint. In order to increase the stability and to reduce the frication of rotations, bearing are also configured. In addition, the interface for knee joint is also designed for connecting the BK leg mechanism. Consequentially, these parts are fabricated and assembled properly. Fig. 4 shows the photo of the hip module in an AK leg mechanism. Note that the upper surface of the AK leg mechanism acts as the interface to directly connect the actuators (*L*1 and *R*7 indicated in Fig. 1) to form a 3 degree of freedoms hip joint.

At the same time, the CAD of the knee part of a BK leg mechanism is shown in Fig. 5. In this figure, a supporting cubic is designed as the frame of one set of tendon pulleys to form the knee joint. In addition, the CAD of the ankle part of a BK leg mechanism is shown in Fig. 6. In this figure, a supporting cubic is designed as the frames of two diagonal tendon pulleys to form the ankle joint. Consequentially, these parts are fabricated and assembled properly. Fig. 7 shows the photo of the knee module in a BK leg mechanism. Meanwhile, Fig. 8 shows the photo of the ankle module in a BK leg mechanism.

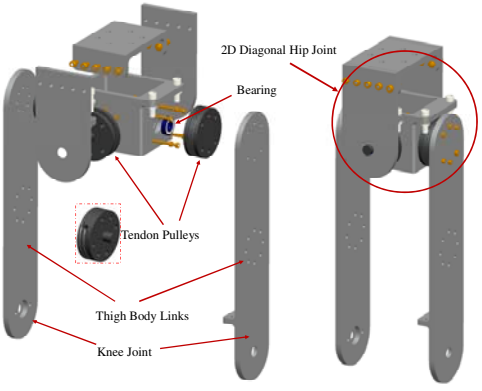


Fig. 3. AK leg mechanism design

Fig. 4. Photo of realized hip module in an AK leg mechanism

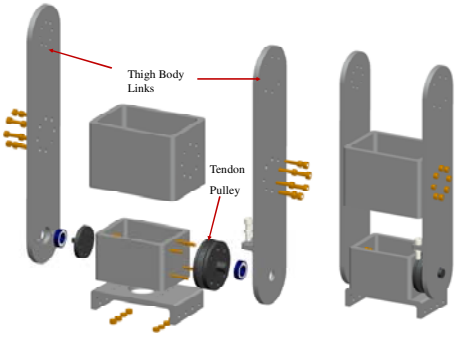


Fig. 5. CAD of the knee module in a BK leg mechanism

In addition to the leg mechanism modules, the actuation module unit is also designed. 12 motor actuators are constructed in this actuation module unit. To eliminate backlashes of motions, 6 harmonic drive gear motors are used. The other motors use conventional spur gear motors to reduce the cost of development. In addition, 10 distal joints are actuated using tendon wires. Two actuators are directed

Fig. 6. CAD of the ankle module in a BK leg mechanism

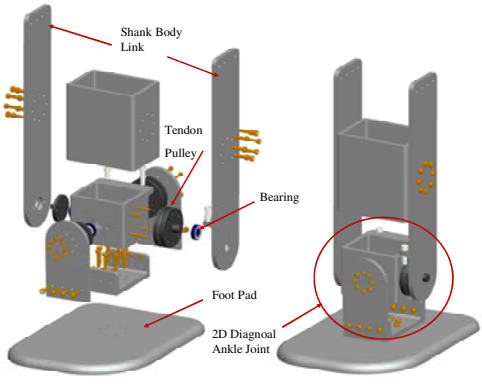


Fig. 7. Photo of realized knee module in a BK leg mechanism

Fig. 8. Photo of realized ankle module in a BK leg mechanism

coupled at the hip joints, and the two actuators also serve as the interfaces of two legs and the actuator module unit. Fig. 9 shows the CAD of the actuator module unit. Fig. 10 shows photos of mechanical parts (left-hand-side) and assembled (right-hand-side) actuation module unit. Note that the tendon pulleys are designed for decoupled motor actuators.

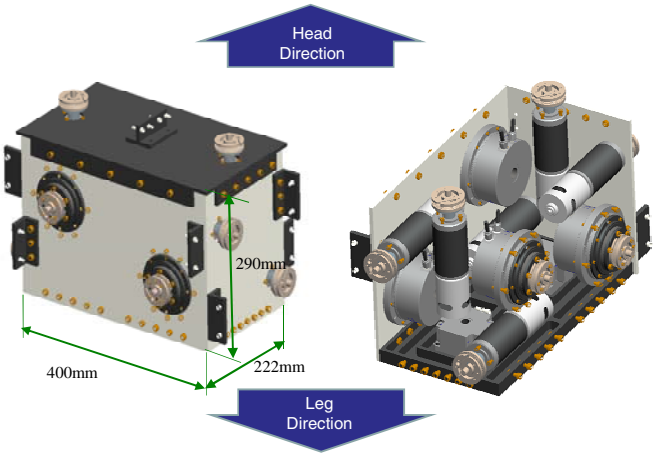


Fig. 9. CAD of actuation module unit

Fig. 10. Mechanical parts (LHS) and assembled (RHS) actuation module unit

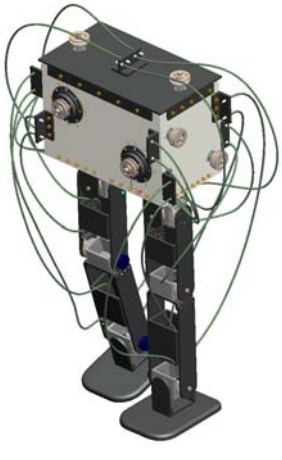


Fig. 11. CAD of the proposed tendon based full size biped humanoid robot

Finally, the whole robot body is properly assembled. Fig. 11 shows the CAD of the proposed tendon based full size biped humanoid robot. At the same time, the assembled physical robot is also shown in Fig. 12. Note that in order to adjust the tendon force of each tendon wire, a screw based tendon adjuster device is desired, and they are shown at the chest (front side of the actuation module unit) of Fig. 12.

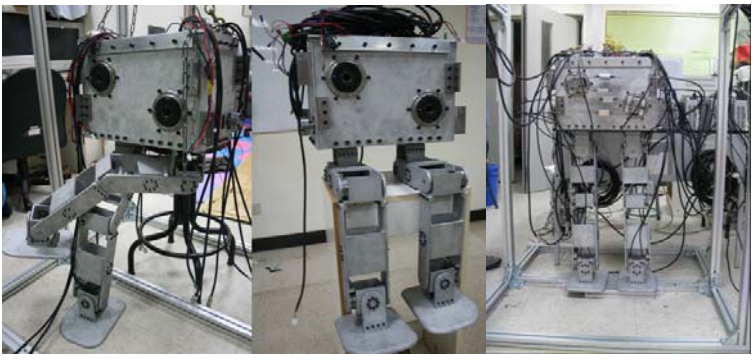


Fig. 12. Photo of assembled physical robot

3 Kinematics

Kinematics is important to model the motions of robots. In this paper, the forward kinematics is constructed based on Denavit-Hartenberg (D-H) representations. The inverse kinematics is established in terms of geometric relationships. The parameters of left leg link appeared in Fig. 13 are shown in Table I. The link parameters of the

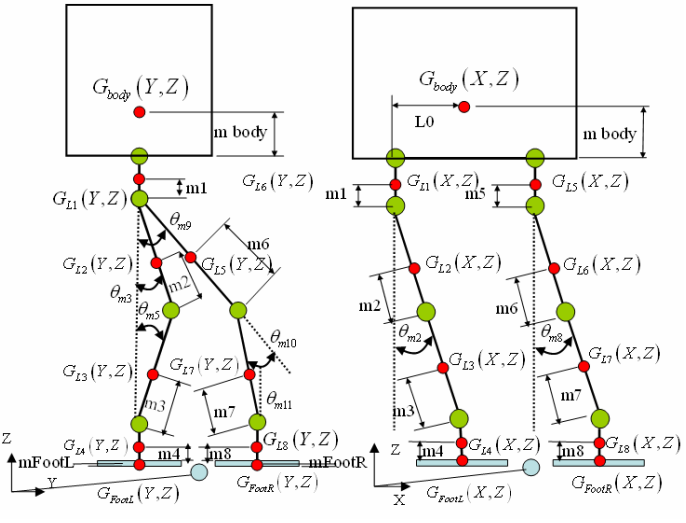


Fig. 13. Definitions of link parameters

Table 1. Robot link parameter measurements

Link Length (mm)		Link Mass (g)		Distance of Link COG (mm)		Initial Angle (deg)	
L_1	85	M_{L1}	878	m_{Body}	95	θ_{m2}	0
L_2	220	M_{L2}	1165	m_1	21	θ_{m3}	0
L_3	292	M_{L3}	1752	m_2	98	θ_{m4}	0
L_4	93	M_{L4}	895	m_3	16	θ_{m5}	0
$Body_W$	402	M_{Body}	33520	m_4	60	θ_{m6}	0
$Body_L$	222	M_{Foot}	1270	m_{Foot}	8	θ_{m8}	0
$Body_H$	290	N/A	N/A	N/A	N/A	θ_{m9}	0
$Foot_W$	180	N/A	N/A	N/A	N/A	θ_{m10}	0
$Foot_L$	240	N/A	N/A	N/A	N/A	θ_{m11}	0
$Foot_H$	15	N/A	N/A	N/A	N/A	θ_{m12}	0
L	107	N/A	N/A	N/A	N/A	N/A	N/A
R	107	N/A	N/A	N/A	N/A	N/A	N/A

right leg can be referred to the left leg because of symmetric relationships. This table will be used for the further COG and ZMP calculations, and it can also be used to generate gait patterns.

3.1 Forward Kinematics

Based on the robot link parameters, the forward kinematics can be defined using the D-H representations. Coordinates of D-H representations is shown in Fig. 14. A D-H table for the left leg is further defined for such a coordinate system, as shown in Table II.

Consequently, the forward kinematics of the left leg is established as shown in Eq. 1. The right leg forward kinematics can also be derived in a similar way, and it is not described here.

$$\begin{bmatrix} P_6^0 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & -a_1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} \cos(\theta_{m1}) & 0 & -\sin(\theta_{m1}) & 0 \\ \sin(\theta_{m1}) & 0 & \cos(\theta_{m1}) & 0 \\ 0 & -1 & 0 & L_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times$$
$$\begin{bmatrix} \cos(90+\theta_{m2}) & 0 & -\sin(90+\theta_{m2}) & 0 \\ \sin(90+\theta_{m2}) & 0 & \cos(90+\theta_{m2}) & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} \cos(\theta_{m3}) & -\sin(\theta_{m3}) & 0 & -L_2\cos(m_3) \\ \sin(\theta_{m3}) & \cos(\theta_{m3}) & 0 & -L_2\sin(m_3) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times$$
$$= \begin{bmatrix} \cos(\theta_{m4}) & -\sin(\theta_{m4}) & 0 & -L_3\cos(\theta_{m4}) \\ \sin(\theta_{m4}) & \cos(\theta_{m4}) & 0 & -L_3\sin(\theta_{m4}) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} \cos(\theta_{m5}) & -\sin(\theta_{m5}) & 0 & \cos(\theta_{m5}) \\ \sin(\theta_{m5}) & \cos(\theta_{m5}) & 0 & \sin(\theta_{m5}) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \tag{1}$$
$$\begin{bmatrix} \cos(\theta_{m6}) & -\sin(\theta_{m6}) & 0 & -L_4\cos(\theta_{m6}) \\ \sin(\theta_{m6}) & \cos(\theta_{m6}) & 0 & -L_4\sin(\theta_{m6}) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

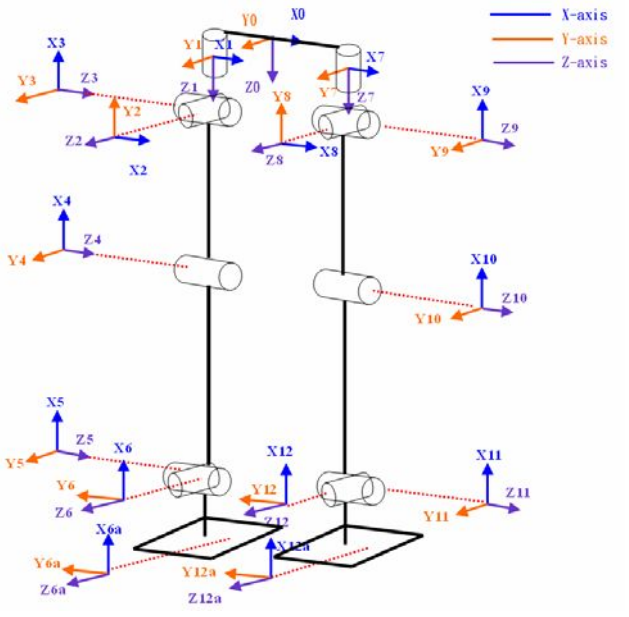


Fig. 14. Coordinates of D-H representations

Table 2. D-H table parameters for left leg

Joint i	α_i (deg)	a_i (mm)	θ_i (deg)	D_i (mm)
$HipL_1$	0	$-L_0$	0	0
$HipL_2$	-90	0	θ_{m1}	L_1
$HipL_3$	90	0	$90+\theta_{m2}$	0
$KneeL_4$	0	$-L_2$	θ_{m3}	0
$AncleL_5$	0	$-L_3$	θ_{m4}	0
$AncleL_6$	-90	0	θ_{m5}	0
$FootL_{6a}$	0	0	θ_{m6}	$-L_4$

3.2 Inverse Kinematics

Inverse kinematics is derived for joint angle control purposes when the path or trajectory of end effectors is planned. In this paper, the geometric relationships are used to derive the inverse kinematics. Fig. 15 shows the geometric relationships for the legs.

By defining trajectories of the hip and ankles, the joint control angles can be derived. Note that θ_{m1} andθ_{m7} are not valuated when the robot walks straightly.

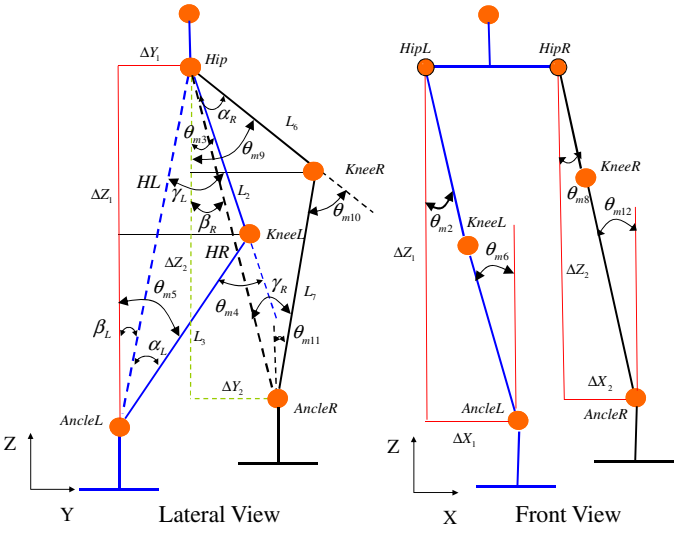


Fig. 15. Geometric relationships for the legs

$$\theta_{m2} = \theta_{m6} = \tan^{-1}\frac{\Delta X_1}{\Delta Z_1} \tag{2}$$

$$\theta_{m8} = \theta_{m12} = \tan^{-1}\frac{\Delta X_2}{\Delta Z_2} \tag{3}$$

$$\theta_{m3} = \cos^{-1}\left(\frac{L_2^2 + HL^2 - L_3^2}{2 * HL * L_2}\right) - \sin^{-1}\left(\frac{\Delta Y_1}{HL}\right) \tag{4}$$

$$\theta_{m4} = \cos^{-1}\left(\frac{L_3^2 + HL^2 - L_2^2}{2 * HL * L_3}\right) + \cos^{-1}\left(\frac{L_2^2 + HL^2 - L_3^2}{2 * HL * L_2}\right) \tag{5}$$

$$\theta_{m5} = \sin^{-1}\left(\frac{\Delta Y_1}{HL}\right) + \cos^{-1}\left(\frac{L_3^2 + HL^2 - L_2^2}{2 * HL * L_3}\right) \tag{6}$$

$$\theta_{m9} = \sin^{-1}\left(\frac{\Delta Y_2}{HR}\right) + \cos^{-1}\left(\frac{L_7^2 + HR^2 - L_6^2}{2 * HR * L_7}\right) \tag{7}$$

$$\theta_{m10} = \cos^{-1}\left(\frac{L_6^2 + HR^2 - L_7^2}{2 * HR * L_6}\right) + \cos^{-1}\left(\frac{L_7^2 + HR^2 - L_6^2}{2 * HR * L_7}\right) \tag{8}$$

$$\theta_{m11} = \cos^{-1}\left(\frac{L_7^2 + HR^2 - L_6^2}{2 * HR * L_7}\right) - \sin^{-1}\left(\frac{\Delta Y_2}{HR}\right) \tag{9}$$

where

$$HR = \sqrt{\left(Y_{HipR} - Y_{AncleR}\right)^2 + \left(Z_{HipR} - Z_{AncleR}\right)^2} \tag{10}$$

$$HL = \sqrt{\left(Y_{HipL} - Y_{AncleL}\right)^2 + \left(Z_{HipL} - Z_{AncleL}\right)^2} \tag{11}$$

$$\Delta X_1 = X_{HipL} - X_{AncleL} \tag{12}$$

$$\Delta X_2 = X_{HipR} - X_{AncleR} \tag{13}$$

$$\Delta Y_1 = Y_{Hip} - Y_{AncleL} \tag{14}$$

$$\Delta Y_2 = Y_{AncleR} - Y_{Hip} \tag{15}$$

$$\Delta Z_1 = Z_{Hip} - Z_{AncleL} \tag{16}$$

$$\Delta Z_2 = Z_{Hip} - Z_{AncleR} \tag{17}$$

4 Kinematics

This paper is a preliminary study of an ongoing project. The gait patterns are simulated based on a given hip and ankle trajectories. Although the ZMP of the robot can be calculated during simulation, the ZMP control is not considered in this paper. The ZMP control will be a future work of this paper. In this paper, a trial-and-error approach is used to justify the parameters of hip and ankle trajectories so that the ZMP is within a safe range of the supporting foot pad. Fig. 16 shows the gait pattern generation approaches by using the trial-and-error methods. Note that it is difficult to find the actual ZMP. However, the distal joint actuation architecture may result in an approximate light leg weight model. Therefore, the ZMP can be proximate using the weight, posture, and acceleration of the COG.

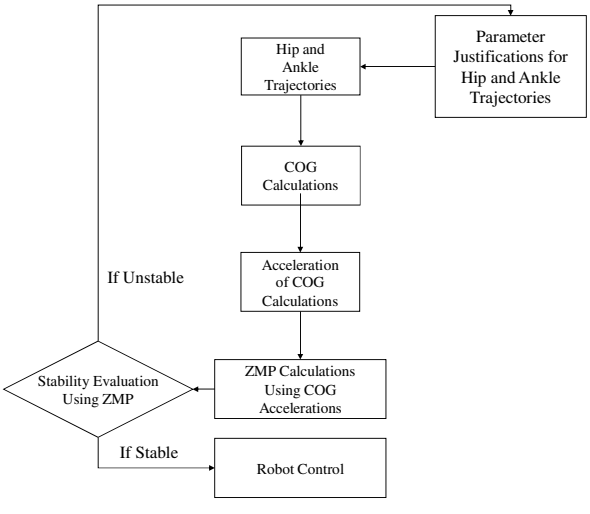


Fig. 16. Pattern generation approaches using the trial-and-error methods

Trajectories of the hip are designed as a sigmoid function to smooth the acceleration curve of hip positions so that the walking stability can be improved. Eq. 18-20 indicates the hip periodic trajectories, where D_x and D_y indicate maximum offsets of the hip in the *x*- and *y*- directions; H is the height of the hip; T_s indicates the time for a gait cycle; t is the time relative to a gait start time; $\alpha$is a constant that adjust the acceleration curve of the hip.

$$X_{Hip}(t) = D_X / \left(\frac{1}{1+e^{(-\alpha * \mathrm{Ts} * \mathrm{t})}} \right) \tag{18}$$

$$Y_{Hip}(t) = D_Y / \left(\frac{1}{1+e^{(-\alpha * \mathrm{Ts} * \mathrm{t})}} \right) \tag{19}$$

$$Z_{Hip}(t) = H \tag{20}$$

At the same time, the trajectories of the ankle are also defined as in Eq. 21-22, where S is the length of a gait cycle in *x*-direction; S_h indicates the maximum height of a gait cycle. Note that X_A is not considered if the robot walks straightly.

$$Y_A(t) = \frac{S/2}{\pi} \left[2\pi \frac{t}{T_s} - \sin\left(2\pi \frac{t}{T_s} \right) \right] \tag{21}$$

$$Z_{\mathrm{A}}(t) = \frac{S_h}{\pi} \left[1 - \cos\left(2\pi \frac{t}{T_s} \right) \right] \tag{22}$$

5 Implementations and Experiments

In this section, the gait patterns are generated based on the approaches of Fig. 16. The gait patterns are simulated using the Matlab software tool. By using trial-and-error approaches, the parameters of hip and ankle are justified to fit stable walking requirements. The simulation conditions are the displacement of the hip is 100 mm in *y*-direction; the gait length in *x*-direction is 300 mm; and the maximum height of the ankle for a gait cycle is 70 mm. Fig. 17 shows the 3D animation segments of the waling simulation.

In addition, the gait pattern data is delivered to an ARM 7 based gait motion controller, as shown in Fig. 19. The gait motion controller uses the gait pattern data to control the robot, and the video segments of motion is shown in Fig. 20. However, due to preliminary studies of this system, differences between simulation and physical robot are not justified. Although the robot may perform the same motion with the simulation, the stability is still a problem for current results.

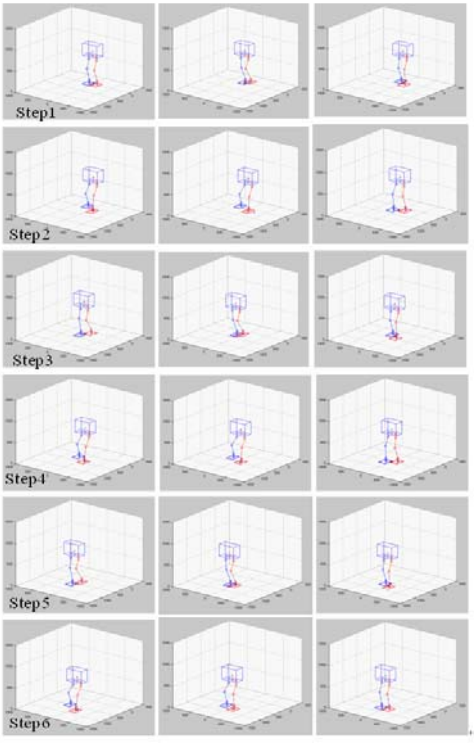


Fig. 17. 3D animation segments of waling simulation using the Matlab

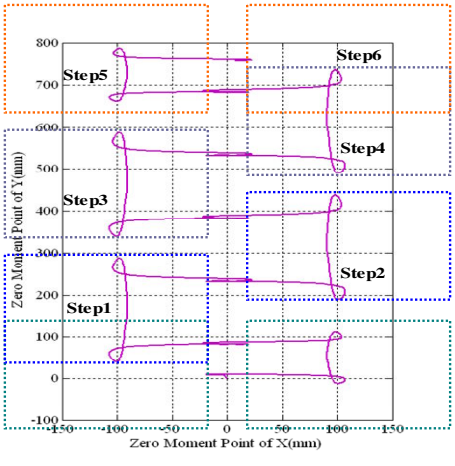


Fig. 18. ZMP trajectory of simulations

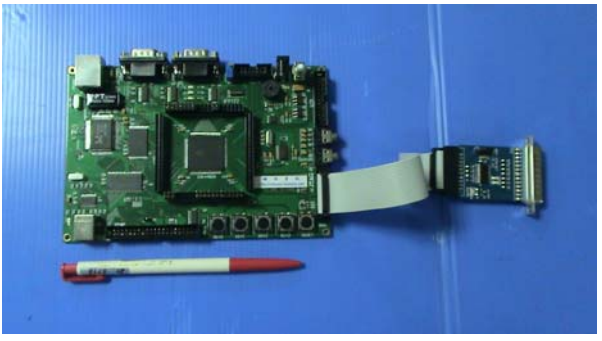


Fig. 19. ARM 7 based gait motion controller

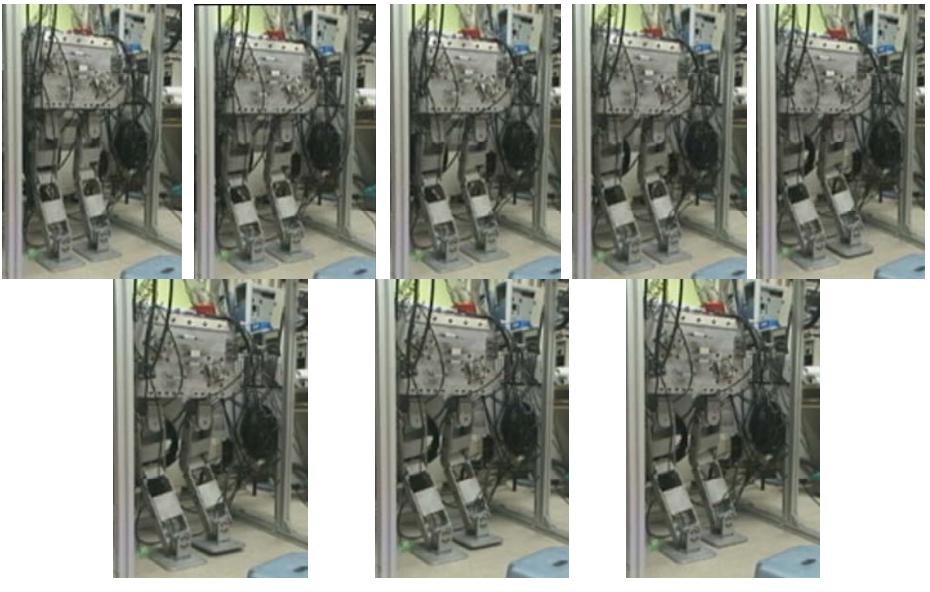


Fig. 20. Video segments of gait motions

6 Conclusion

This paper presents a tendon based full size biped humanoid robot walking platform. The mechanical structure is constructed based on distal joint actuations. Therefore, the features of wiring efforts, COG calculation accuracy, and ground contact impact reductions can be desired. In addition, the gait patterns generations are developed based on trial-and-error approaches by adjusting the hip and ankle trajectories to achieve the stability of walking. This paper is an ongoing project, and results of this paper just perform a preliminary study of our work. In the future, the simulation parameters will be justified according to the walking platform prototype to achieve stable walking. At the same time, the ZMP control is also developed to perform autonomous walking.

Acknowledgment

This work is supported by National Science Council under Grant number 95-2221-E-011-226-MY3.

References

1. Ogura, Y., Aikawa, H., Shimomura, K., Kondo, H., Morishima, A., Lim, H.O., Takanishi, A.: Development of a new humanoid robot WABIAN-2. In: IEEE International Conference on Robotics and Automation, pp. 76–81 (2006)
2. Sakagami, Y., Watanabe, R., Aoyama, C., Matsunaga, S., Higaki, N., Fujimura, K.: The intelligent ASIMO: system overview and integration. In: IEEE/RSJ International Conference on Intelligent Robots and System, vol. 3, pp. 2478–2483 (2002)
3. Sanada, H., Yoshida, E., Yokoi, K.: Passing under obstacles with humanoid robots. In: IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 4028–4034 (2007)
4. Huang, Q., Yokoi, K.: Planning walking patterns for a biped robot. IEEE Trans. on Robotics and Automation 17, 280–289 (2001)

FPGA-Based Vocabulary Recognition Module for Humanoid Robot

Yu-Te Su, Chun-Yang Hu, and Tzuu-Hseng S. Li

aiRobots Laboratory, Department of Electrical Engineering
National Cheng Kung University, Tainan 70101, Taiwan, R.O.C.
N2895109@mail.ncku.edu.tw, thsli@mail.ncku.edu.tw

Abstract. This paper focuses on using FPGA board to realize the ability of vocabulary recognition for humanoid robot. At first, the system structure of the humanoid robot system is setup. The central process unit of the humanoid robot is a FPGA board, ALTERA Nios II EP2C20F324C8, which not only controls all the motors of robot but also processes the information of vision system. The vocabulary recognition method is then introduced. We apply the image segment to find the valid region, and use the encoding method to sample the word. After matching algorithm, we use a speech module, Emic TTS module, to pronounce the word. Finally the experiments verify the procedure of the proposed module and demonstrate the feasibility of the vocabulary recognition and speak out function for the humanoid robot.

Keywords: FPGA-based, humanoid robot.

1 Introduction

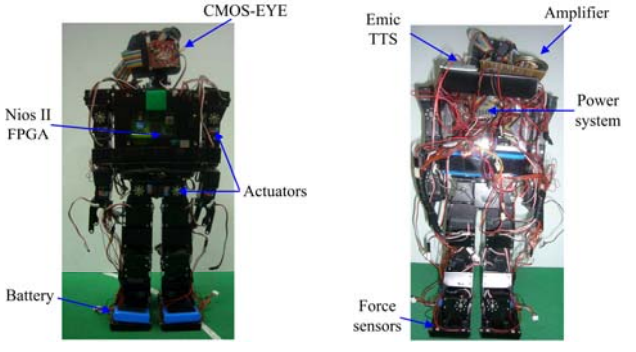
The humanoid robot mentioned in this paper is called aiRobot-2. The function of aiRobot-2 can be regarded as two mainly parts in general: the balance motion control [1-5] and the vision system [6, 7]. We utilize accelerometer sensors and force sensors equipped on aiRobot-2 to calculate the ZMP of the robot and adjust the actuators to stabilize the motion. The issue of the stability has been discussed in the literature [1-4]. We focus on the intelligence of aiRobot-2 with the vision system and the speech system [8].

This system architecture of a humanoid robot takes the major of the cost. For example, the robot uses the PDA (Personal Digital Assistance) or other computation systems to be the vision system, and uses the embedded controller or PC to be the motion control system, like ASIMO [9]. In order to reduce the cost, aiRobot-2 combines two computation systems in one FPGA embedded system in this paper. In general, the humanoid robot should have the abilities of recognition for specific purpose and even interact with human [10]. Therefore, the aiRobot-2 is designed even to recognize the vocabularies, execute some mathematical calculations and pronounce them correctly. The detail information of vision system and speech system will be introduced by following sections.

J.-H. Kim et al. (Eds.): FIRA 2009, LNCS 5744, pp. 151–160, 2009.

2 The System Structure of the Humanoid Robot

The hardware of the small-sized humanoid robot includes: twenty nine servomotors, a CMOS sensor, eight pressure sensors, an accelerometer, four Li-Po batteries, Nios II evaluation board, and an integrated circuit board. The photos of aiRobot-2 are shown in Fig. 1. Table 1 shows the basic specification of the humanoid robot. All the information of the sensors is delivered to the Nios II FPGA board. After calculating by the CPU, the actuators act according to the information result computed by the FPGA board. We introduce the hardware and software briefly via three sub-sections including center process unit, the vision system and the speech module.



(a) The complete structure of aiRobot-2 (b) The disposition of motors on aiRobot-2

Fig. 1. The mechanism design of aiRobot-2

Table 1. The specification of our robot

Height	50cm
Width	20cm
Depth	10cm
Weight	3kg
Degree of Freedom	29
Central Process Unit	NIOS II 2C20 FPGA Board

2.1 Central Process Unit

The core of aiRobot-2 is the ALTERA Nios II EP2C20F324C8 FPGA board used as the central process unit. Nios II is a mulCentral process unit Vti-function FPGA board and the operating interface of the robot is built by the ALTERA SOPC Builder. The hardware architecture is established by the Quartus II software. We use the SOPC Builder to design the CPU module and other devices such as CMOS image sensor signal controller, sensors and motors. Additionally, the functions in the blocks of the peripheral devices are written in Verilog HDL. The control strategy and image

algorithm are written via the C/C++ language in the ALTERA development platform, the Nios II Integrated Development Environment (IDE). Therefore, the functions of the robot can be operated by downloading the software into the Nios II evaluation board through the JTAG debug interface.

2.2 Vision Sensor

We use CMOS-EYE image sensor module manufactured by ASAKUSAGIKEN Company in Japan for the eye of our humanoid robot. This device is indeed small and light. If the vision sensor is too heavy or too large at the head of aiRobot-2, it will increase the difficult to rotate the head. Because we process the image by the NIOS II FPGA Board which has no display device to show the image, we develop a human-machine interface with Borland C++ Builder (BCB) software besides the Nios II IDE. The algorithm of object recognition and vocabulary recognition are all written in the Nios II IDE interface. The BCB interface is used to set parameters of objects, show the image and the coordinate information. It is more convenient for user to write the program and debug by combining the Nios II IDE with the BCB interface.

2.3 Speech Module

In order to let aiRobot-2 interact with human beings by speech, we utilize the Emic TTS (Text-To-Speech) Module manufactured by Parallex Company in United States. The the module is shown in Fig. 2. The Emic TTS module proceeds command control and data transportation by TTL (Transistor-Transistor Logic) level UART (Universal. Asynchronous Receiver and Transmitter) shown in Fig. 3. We send the ASCII commands to control the function including the volume, the speed and the pitch. Then the Nios II sends the ASCII data recognized by the vocabulary recognition algorithm to the Emic TTS module to say the vocabulary. Because the power of the speaker driver on the Emic TTS is only 300mW, we design an audio amplifier circuit board with LM386N-3 IC and connect an 8Ω 500mW speaker to amplify the audio volume from the Emic TTS shown in Fig. 4. The advantages of this module are small and simple to use. LM386N-3 is a low voltage audio power amplifier. The default gain of the IC is 20dB. The gain can be adjusted by the capacity between the pin1 and pin8, and its maximum value is 200 dB.

Fig. 2. Emic Text-To-Speech module

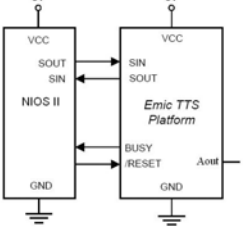


Fig. 3. Communication with the host device

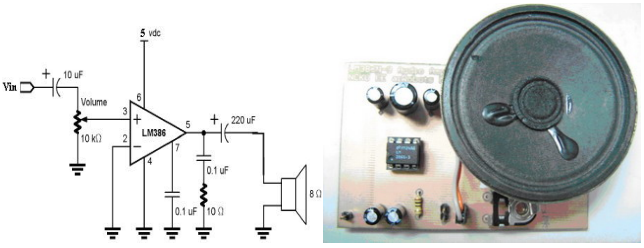


Fig. 4. The figure of the LM386N-3 amplification circuit

3 Vocabulary Recognition

If the humanoid robot can only execute tasks unilaterally and can not interact with human, it loses the charms to human. Take account of the market to the teaching robot. If the children can be educated by the humanoid robot, they may be interested in the humanoid robot. Furthermore, we can achieve the objective that aiRobot-2 interacts with human. On the basis of above reasons, we design two events including the English vocabularies recognition and numbers comparison to show the teaching functions with aiRobot-2. The steps of recognition can be divided into the vision system and the control strategy system as shown in Fig. 5. Each step is addressed in detail as follows.

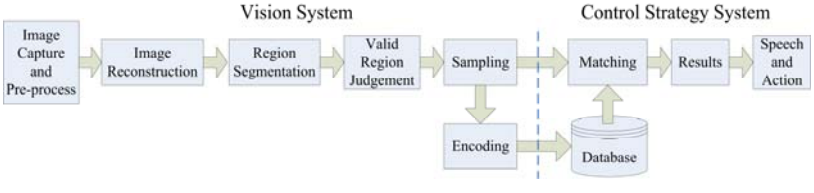
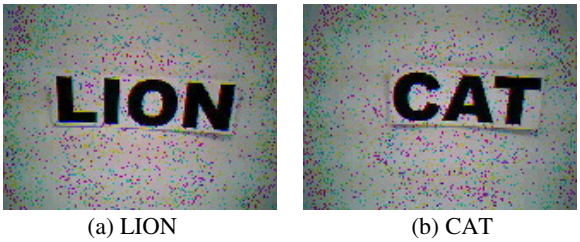


Fig. 5. The flowchart of the recognition

3.1 Image Capture and Pre-process

The image is also captured by the CMOS-EYE and delivered to Nios II as Fig. 6. After we got the image, we need to pre-process the image. Firstly, we have to filter



(a) LION (b) CAT

Fig. 6. The original images for recognition

Fig. 7. The threshold images for "LION" and "CAT"

out the noises, and convert the color image to the gray level image. Finally, we convert the gray level image to the binary image as Fig. 7. In order to speed up the processing, the background in our experiments is white. Thus, we can reduce the unnecessary process of color filtering.

3.2 Image Reconstruction

The binary image which we obtain from above steps may contain the broken gaps. We have to reconstruct these gaps. The common methods for image reconstruction in the morphology are dilation and erosion [11-13]. Another advantage for the process is to filter out the isolated dot. We are going to discuss these two methods below.

The dilation operation is to find the maximum value in the structuring element. The maximum value is also the result of the dilation operation. The symbol $\oplus$ in (1) represents the dilation operator.

$$A \oplus B(x, y) = \max\{A(x-i, y-j) \mid (x, y) \in A(i, j) \in B\} \tag{1}$$

$$A \Theta B(x, y) = \min\{A(x+i, y+j) \mid (x, y) \in A(i, j) \in B\} \tag{2}$$

where A is the original image, B is the structuring element, (x, y) is the coordinate of the pixel in the structuring element, and (i, j) is the coordinate of the pixel in the original image.

On the other hand, the erosion operation is to find the minimum value in the structuring element. The symbol Θ in (2) represents the erosion operator. In the programming, we use the logic operators "|" and "&" to substitute the maximum and minimum calculation to reduce the computation, respectively. Fig. 8 shows the result of the dilation operation. We can see that isolated dots around the word are also be dilated simultaneously. However, the isolated dots disappeared with the erosion

Fig. 8. The dilation images "LION" and "CAT"

Fig. 9. The erosion images "LION" and "CAT"

Fig. 10. The combination images "LION" and "CAT"

operation in Fig. 9 and the characters are also rough. To compensate the image and reach the effect, we combine these two operations by performing erosion once and then dilation once for better performance as shown in Fig. 10. All the structuring elements used above are 3×3 .

3.3 Region Segmentation

The purpose of the region segmentation is to separate the different information [14-15]. We preserve the information we needed and filter out the others. The method also cuts down the amounts of the data and improves the speed of the system. The purpose of the region segmentation in our experiment is to discriminate each possible character from the background. The explanation of the method is shown in Fig. 11. We scan the binary image column by column from left to right. Once we detect the character pixel, we record the previous column of the detected column called the character start point. Continue scanning until there is no character pixel on the whole

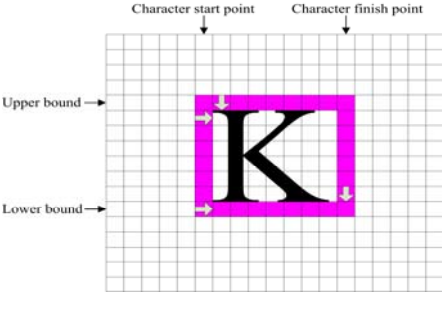
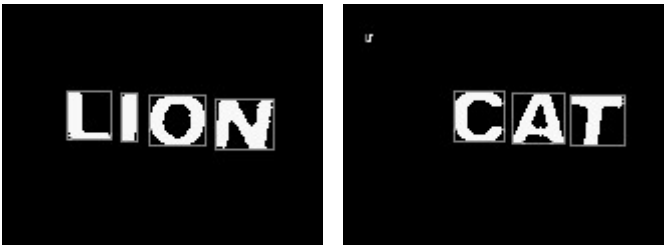


Fig. 11 The explanation of the Region segmentation



(a)LION (b) CAT

Fig. 12. The results of the region segmentation

column. We record the next column of the detected column called the character finish point. So far, one possible character region searching is accomplished. Then, we keep on scanning till last column. Similarly, we can locate the up and lower bound for each character by using this method. The results are shown in Fig. 12.

3.4 Valid Region Judgment

After each possible character region found in the image, we are going to judge which is the proper character region. We set a threshold according to the length and the width in our experiments. Adjudge the boundary of each possible region that satisfies the threshold or not. If the possible region satisfies the threshold, we judge the one is the proper region. Others which don't satisfy the threshold are not being considered to reduce the computation and increase the accuracy of the recognition.

3.5 Sampling and Encoding

Before practicing recognition, we need to establish a database for matching up reference. We segment the first proper character region into 25 grids of 5×5 as in Fig.13. Calculate the area of each grid and the feature pixel number in each grid. Then, we figure out the ratio $(r_i, i = 1, 2, \cdots, 25)$ with the feature pixel number in each grid divided by the area of each grid. The ratio which is encoded for four levels with 0,1,2,3 as (3) is the code $(e_i, i = 1, 2, \cdots, 25)$ and the code is saved to the corresponding

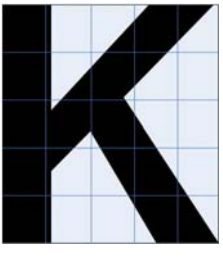


Fig. 13. The 5×5 segment region

position of the grid. After practicing the procedure above, we finish the sampling and the encoding of one character. We have to establish the template of every possible character into the matrix of the database to increase the accuracy of the recognition. So far, we have established 26 English letters and ten numbers including 0~9.

$$e_i = \begin{cases} 0 \quad , & 0 \le r_i \le \frac{1}{5} \\ 1 \quad , & \frac{1}{5} < r_i \le \frac{1}{2} \\ 2 \quad , & \frac{1}{2} < r_i \le \frac{4}{5} \\ 3 \quad , & \frac{4}{5} < r_i \le 1 \end{cases} \tag{3}$$

3.6 Match Up

We match up the proper character region and each template in the database. The procedure is to calculate the Euclidean distance between the sampling code of the proper character region and each template in the database as

$$d = \sqrt{\sum_{i=1}^{25} (x_i - t_i)^2} \tag{4}$$

where d denotes the Euclidean distance, x_i denotes the sampling code of the proper character region and t_i denotes the template in the database. If the d is zero, we choose this template character to be the result and break out this procedure. Otherwise, we choose the template which has the shortest Euclidean distance to be the result. The shortest distance means the least error. If there are rest proper character regions, keep on the matching until last one.

4 Experiment Result

The results of the vocabulary recognition are shown in Fig. 14. The vocabulary "LION" is that aiRobot-2 needs to recognize. Firstly, we put the vocabulary card on the board in front of the robot as in Fig. 14(a). Then aiRobot-2 is going to recognize this word. After finishing the identification of the vocabulary, aiRobot-2 speaks out the word "LION" which are shown in Fig. 14(b)-(c). Fig. 14(d)-(g) are the original image captured from the CMOS-EYE and the results of the recognition algorithm which display on the lower and upper picture of the BCB interface, respectively. Finally, the recognition result is shown on the Nios II IDE as in Fig. 14(h). The first line is the region segmentation block. The second line is the result of the recognition. The last line is the valid segmentation block.

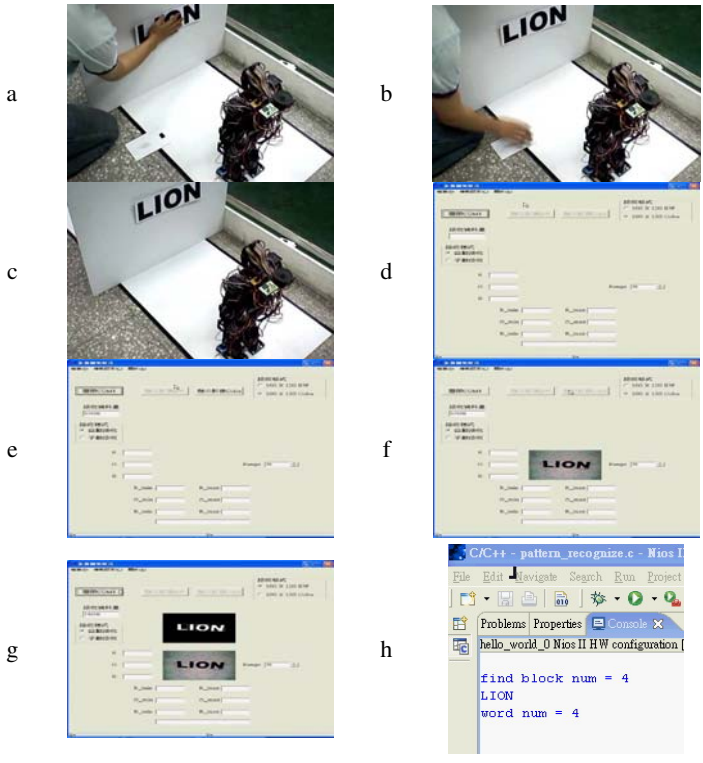


Fig. 14. The strategy of the vocabulary recognition

5 Conclusion

Development of a FPGA-based vocabulary and speech modules for a small-sized humanoid robot, aiRobot-2, is presented in this paper. All the image processing and control system are operated on one FPGA board without other computation system. The overall system is an FPGA visual feedback system. The main algorithm of the image processing system is vocabulary recognition. The result of this algorithm also has been examined. Finally, the experiment results show the efficiency and validity of vocabulary recognition and speech function for a humanoid robot.

Acknowledgment

This work was supported by National Science Council of Taiwan, R.O.C, under Grant NSC95-2221-E006-0-382-MY3.

References

1. Hsu, C.-L., T.-H.S. Li, et al.: Design and Implementation of an SOPC-Based Small Sized Humanoid Robot. In: 2007 CACS Int. Automatic Control Conference, Taiwan (November 2007)
2. Li, T.-H.S., Su, Y.-T., et al.: Stair-Climbing Control of Humanoid Robot using Force and Accelerometer Sensors. In: Proc. of SICE 2007, Japan (September 2007)
3. Su, Y.-T., Li, T.-H.S., et al.: Design and Implementation of Fuzzy Auto-Balance Control for Humanoid Robot. In: 2007 FIRA RoboWorld Congress, USA (June 2007)
4. Li, T.-H.S., Hui, J.-J., et al.: Improvement and Development of Humanoid Soccer Robot NCKU-II. In: 2006 FIRA RoboWorld Congress, German (July 2006)
5. Li, T.-H.S., Hsu, C.-L., Hu, C.-Y., Su, Y.-T., Lu, M.-F., Liu, S.-H.: SOPC Based Weight Lifting Control Design for Small-Sized Humanoid Robot. In: Proc. ICCAS 2008, Korea (October 2008)
6. Su, Y.-T., Li, T.-H.S., et al.: Omni-Directional Vision-Based Control Strategy for Humanoid Soccer Robot. In: Proc. of 2007 IEEE IECON (November 2007)
7. Su, Y.-T., Hu, C.-Y., Lu, M.F., Chang, C.-M., Lai, S.-W., Liu, S.-H., Li, T.-H.S.: Design and implementation of SOPC based image and control system for HuroCup. J. of Harbin Institute Tech (New Series) 15, 41–46 (2008)
8. Lund, H.H.: Modern artificial intelligence for human-robot interaction. Proc. IEEE 92(11), 1821–1838 (2004)
9. Adams, J.A., Keskinpala, H.K.: Analysis of perceived workload when using a PDA for mobile robot teleoperation. In: Proc. IEEE Int. Conf. on Robotics and Automation, April 2004, vol. 4, pp. 4128–4133 (2004)
10. Ido, J., Matsumoto, Y., Ogasawara, T., Nisimura, R.: Humanoid with interaction ability using vision and speech information. In: Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, October 2006, pp. 1316–1321 (2006)
11. Su, C., Haralick, R.M.: Recursive erosion, dilation, opening, and closing transforms. IEEE Transactions on Image Processing 4(3), 335–345 (1995)
12. Gil, J.Y., Kimmel, R.: Efficient dilation, erosion, opening, and closing algorithms. IEEE Transactions on Pattern Analysis and Machine Intelligence 24(12), 1606–1617 (2002)
13. Bödvarssona, B., Klima, S., Mørkebjerga, M., Mortensena, S., Yoona, C.H., Chena, J., Maclarena, J.R., Lutherb, P.K., Squirec, J.M., Bonesa, P.J., Millane, R.P.: A morphological image processing method for locating myosin filaments in muscle electron micrographs. Image and Vision Computing 26(8), 1073–1080 (2008)
14. Gevers, T.: Robust Segmentation and Tracking of Colored Objects in Video. IEEE Transactions on Circuits and Systems for Video Technology 14(6), 776–781 (2004)
15. Casey, R.G., Lecolinet, E.: A survey of methods and strategies in character segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence 18(7), 690–706 (1996)

Motivation and Context-Based Multi-Robot Architecture for Dynamic Task, Role and Behavior Selections

Dong-Hyun Lee and Jong-Hwan Kim

Department of Electrical Engineering and Computer Science
KAIST
Daejeon, Korea
{dhlee,johkim}@rit.kaist.ac.kr
http://rit.kaist.ac.kr

Abstract. This paper proposes a multi-robot coordination architecture for dynamic task, role and behavior selections. The proposed architecture employs the motivation of task, the utility of role, a probabilistic behavior selection and a team strategy for efficient multi-robot coordination. Multiple robots in a team can coordinate with each other by selecting appropriate task, role and behavior in adversarial and dynamic environment. The effectiveness of the proposed architecture is demonstrated in dynamic environment robot soccer by carrying out computer simulation and real environment.

Keywords: Multi-robot coordination, probabilistic behavior selection, team strategy.

1 Introduction

Many practical applications such as reconnaissance and surveillance can be accomplished more effectively and efficiently by using a team of robots rather than using just a single robot to save both time and effort. There might be more than just one task that need to be accomplished in such applications, where each task can be completed by robots with different roles. Consequently, robots in a team are required to select their task, role and behavior in an appropriate manner. Especially in adversarial and dynamic environment, poor task, role and behavior selections may cause tasks more difficult or even impossible to be completed. The task in this paper requires loosely-coupled coordination such as multi-robot object tracking, reconnaissance and exploration. The robots that perform the same task have a similar level of capability, possibly with different resources. There were behavior-based approaches ([1]-[2]) and market-based approach to deal with such problems ([3]-[6]).

This paper proposes a multi-robot coordination architecture, called MCMRA (Motivation and Context-based Multi-robot Architecture), which deals with task, role and behavior selections issues. The proposed architecture uses the

J.-H. Kim et al. (Eds.): FIRA 2009, LNCS 5744, pp. 161–170, 2009.

motivation of a task, which considers robot's capability, and the required capability for the task to maximize the efficiency of resource usages. To select a task, each robot calculates the motivation of each task, generates a context information which consists of environmental state and other robot's condition, and calculates a task strategy ratio which controls the weight of each task based on the strategy and environmental state. After the task selection, one of the roles that belongs to the selected task is selected based on the priority among roles, current role of other robots and the utility of each role. The confabulation method is used for behavior selection, which considers the robot's task, role and environmental state [7][8]. The problem which the architecture deals with can be categorized as a *ST-SR* (Single-Task robots and Single-Robot tasks) problem, where each robot is capable of executing at most one task at a time and some tasks can require multiple robots, in terms of the taxonomy of multi-robot task allocation [9]. The proposed architecture is applied to the game of robot soccer [10].

The rest of this paper is organized as follows. Section II explains the proposed architecture, MCMRA, in detail. In section III, results are described in detail. Concluding remarks follow in Section IV.

2 MCMRA

Motivation and Context-based Multirobot Architecture (MCMRA) considers a motivation of robot for each task, a context of environment and other robots to select its task, role and behavior properly in a distributed manner. The architecture is shown in Fig. 1.

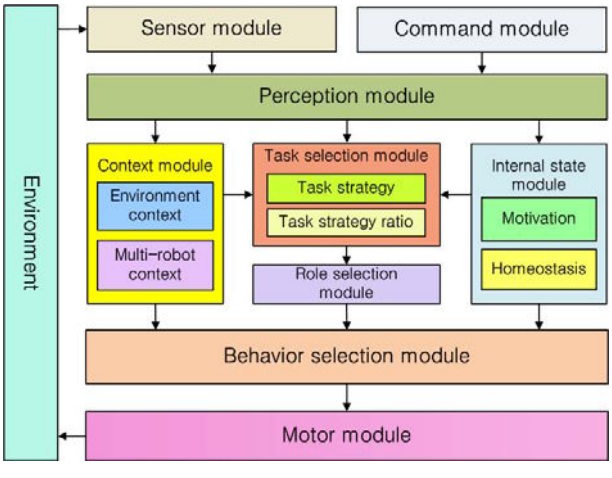


Fig. 1. MCMRA

The sensor module senses the environment and the command module receives commands from user. The perception module generates perception information from sensor data. The internal state module consists of motivation and homeostasis. The context module consists of the environment context and the multi-robot context. The task, role and behavior selection modules are introduced. The Detailed description of each module is provided in the following.

2.1 Internal State Module

The internal state module consists of motivation and homeostasis.

Motivation. The motivation represents a desire to do a task. It is calculated by the capability of the robot and the cost of performing the task. The motivation of robot i $(i = 1, ..., n)$ on the task j $(j = 1, ..., s)$ can be defined as

$$m_i^j = f_m(C_i, P_i, T^j) \tag{1}$$

where C_i is the robot capability in a vector form, which represents the robot's ability such as localization and box pushing. P_i is the perception vector such as the distance from a target and the number of objects around the robot. T^j is the requirement vector of task j and represents what kind of capabilities the task requires and how the cost for the task is estimated. Robot can calculate the degree of its ability on the task from C_i and T^j and the estimated cost on the task from P_i and T^j.

By utilizing motivation, robot can inform the other robots what kind of tasks they can do better considering its capability and the estimated cost for the task. Each robot considers information from the others as well as its own motivation when the task is selected.

Homeostasis. The meaning of homeostasis is the ability or tendency of an organism or cell to maintain internal equilibrium by adjusting its physiological processes. This concept was applied to artificial robots to express the physical condition such as desire to sleep, eat and evacuate [11]. Homeostasis in this paper is used to keep robot in a stable condition when a task is performed. It can be used to check the malfunction of hardware resources such as sensor and actuator. This can increase the robustness of multi-robot coordination.

2.2 Context Module

In distributed multi-robot system, each robot should be able to select its own task and role by itself. For this, the information such as environment and other robot's state should be shared among robots in a team not to compete each other and to perform an efficient coordination. Thus, the environment context module and the multi-robot context module are provided for information sharing.

Environment Context Module. The environment context module combines the local information from the robot's perception module and the other robot's perception from the multi-robot context to estimate the environmental state of robot team. It does not necessarily mean that whole perception data of robots in a team should be shared.

Multi-Robot Context Module. In MCMRA, robots broadcast their motivations on tasks, position and messages such as *Help* and *Abandon* and they record the received information in the multi-robot context. By doing this, all robots in a team can have the identical multi-robot context and each of them can select a task without any conflict by using the multi-robot context and the same task and role selections algorithm. When a robot found a problem with its homeostasis, it sends *Abandon* message to others. *Help* message is also used when the robot requires a help from other robots.

2.3 Task Selection Module

Task selection module selects a task based on the internal state of the robot, task strategy, task strategy ratio of the team, the multi-robot context and messages from other robots.

Task Strategy and Task Strategy Ratio. The task strategy is a set of weights of tasks. The more weights a task gets, the more robots can select the task. The weights in a task strategy are predefined by user. In case of robot soccer, there may be two tasks, offense and defense and the user can make offensive strategy in which the weight of the offense task is selected to be higher than the one of defense task. The *l-th* task strategy, str_l is defined as follows:

$$str_l = [w_1, w_2, ..., w_s] \ (l = 1, ..., e) \tag{2}$$

$$\sum_{k=1}^{s} w_k = 1 \qquad (0 \leq w_k \leq 1)$$

where w_k is the weight of $task_k$. Total sum of the weights are one so that the weight can express the relative proportion among tasks.

Once the task strategy is decided, all tasks can get weights and then the task strategy ratio adjusts the weights of tasks based on the environmental changes. For example, consider there are two areas A and B in Mars and the team of robots explores the areas to find an ice. The user first selects the *strategy 1* which puts higher weight on area A to make robots to selects area A rather than B. During the exploration, the robot which had explored the area B finds the traces of the river and broadcasts the information to other robots. Then the strategy ratio of all robots in a team adjusts the weights of the tasks to make the weight of the task B higher than that of task A.

By using the strategy, the user can decide the weights of tasks based on the user's knowledge and the robots can efficiently adjust the weights of tasks by using the strategy ratio.

Task Selection. The first step of task selection is to check the homeostasis of robot to decide whether it is available for performing tasks. The second step is to check if there is any robot which requested a help by checking *Help* message from multi-robot context. The final step is to select a task based on the motivation, the number of allocated robots for the task and the other robot's motivations in the multi-robot context.

2.4 Role Selection Module

A task is composed of several roles and each role is performed by a robot to accomplish the task. Each role in a task has different priority based on the relative importance among roles so that the role with higher priority should be selected by robots prior to other roles with lower priority. Robots with the same task should consider each other to select a role without conflict. As we mentioned in Section I, robots with the same task have similar capabilities. Thus, the role selection algorithm does not consider the hardware capability of each robot, instead it considers the utility of each robot on each role. The utility of robot i on role k $(j = 1, ..., v)$ in task j can be defined as

$$u_i^{jk} = f_u(Z_i, V^{jk}) \tag{3}$$

where Z_i is the vector of robot information such as the location of robot and the distance from a target. Z_i can be obtained from the multi-robot context. Note that each robot can calculate the other robot's utility by using the multi-robot context. V^{jk} is the role information vector such as the location or the area that the role covers.

2.5 Behavior Selection Module

The behavior selection module uses confabulation method [7]. Suppose that there are assumed facts, α and β and the conclusion, ϵ. This method assumes that the maximization of cogency $p(\alpha\beta \mid \epsilon)$ is equivalent to the maximized product of $p(\alpha \mid \epsilon) \cdot p(\beta \mid \epsilon)$. That is, if sensor information is used as the assumed fact and robot's behavior is considered as the conclusion, then the behavior that maximizes the probability can be considered as the most suitable behavior of a robot in the situation. This method assumed that all probabilities $p(j \mid l)$ between symbols j and l are known. The confabulation method was proposed for selecting the behavior of software robot [8].

The task, role, internal state and the information from context are used as assumed facts and the robot's behavior as the conclusion. The behavior of robot i on the $role_k$ in $task_j$ is defined as

$$beh_i^{jk} = \arg\max_{\epsilon}(p(\alpha \mid \epsilon) \cdot p(\beta \mid \epsilon) \cdot p(\gamma \mid \epsilon) \cdot p(\delta \mid \epsilon)) \tag{4}$$

where $\alpha, \beta, \gamma, \delta$ and ϵ are the symbolized values of $task_j$, $role_k$, environment context, perception and beh_i^{jk}, respectively.

3 Experiments

The proposed architecture was applied to the robot soccer domain. The robot soccer has two tasks, offense and defense tasks. The offense task has five roles, *Striker*, *Forward*, *CenterWing*, *LeftWing* and *RightWing* and the defense task also has five roles, *Goalkeeper*, *Sweeper*, *CenterBack*, *LeftBack* and *RightBack*.

There are eight behaviors, *Pass*, *Shoot*, *CatchBall*, *Penetration*, *Backup*, *intercept*, *GoRoleArea* and *Wait*. Two kinds of robots were defined, offensive and defensive robots, by their capability. The offensive robot can move faster than defensive one and has lighter and slender frame structure. The defensive robot, on the other hand, moves slower but heavier and wider frame structure can make the robot perform efficient defense. The motivation of robot for each task was calculated by the robot's capability on the task and the distance from the opponent team's goal. The utility of each role was calculated by the robot's location on the playground and the distance from a ball.

Task selection with different task strategy. This simulation demonstrates the task selection of robots in a team with different task strategy and environmental changes.

Three strategies, general, offensive and defensive strategies were defined. The general strategy puts the same weights on both offense and defense task, respectively. The offensive and defensive strategies put more weight on offense and defense task respectively. After the strategy is decided, the weights can be changed by the task strategy ratio which considers environmental changes. The time ratio ($\frac{current\ time}{full\ time}$), the team which got a ball (three conditions of the ball: free ball, home team ball, opponent team ball) and the score ratio ($\frac{Home\ team\ score - Opponent\ team\ score}{Max\ score}$) were considered as the environmental changes in robot soccer domain.

In the simulation, a team consisted of total six robots which consist of three offensive robots and the other three defensive ones. The offensive robot has higher capability on offense task than defense task and the defensive robot has higher

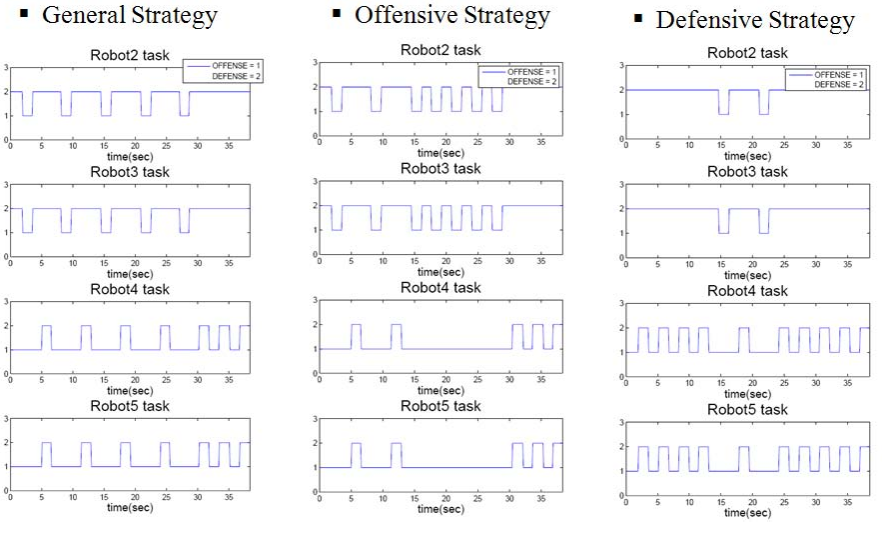
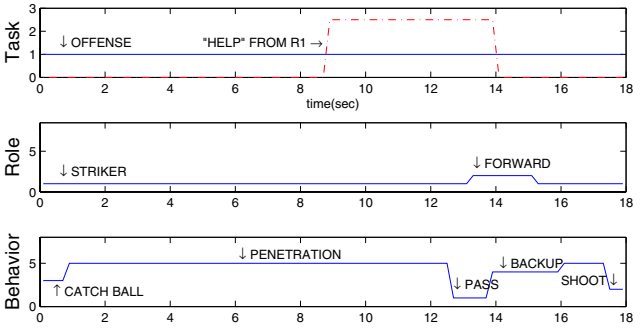
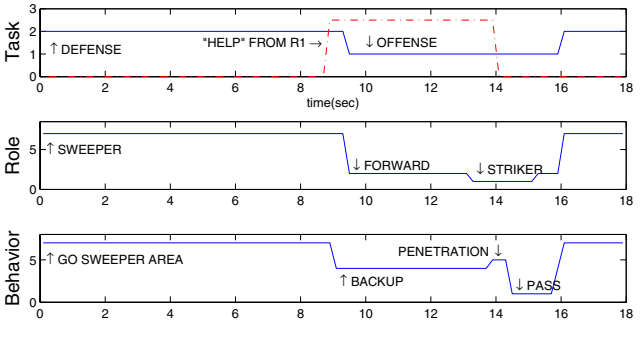


Fig. 2. Task Selection on Different Strategies



(a)



(b)

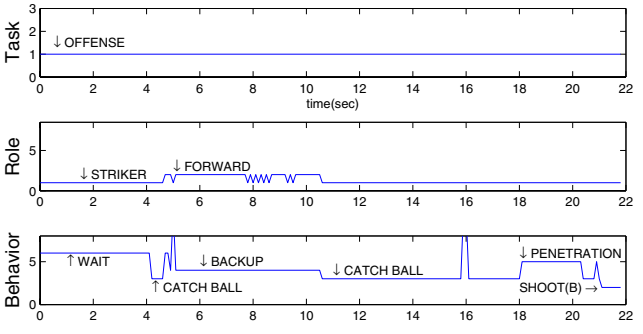
Fig. 3. Task, Role and Behavior selections. (a) $robot_1^G$, (b) $robot_3^G$.

capability on defense task than offense task. The strategy was selected before the simulation. The predefined environmental changes were applied to the team. The time ratio was increased from 0 to 1 until the end of the play and the ball owner was changed like Free ball ⇒ Home team ball ⇒ Free ball ⇒ Opponent team ball. The score ratio was changed; 0:0 ⇒ 0:1 ⇒ 0:2 ⇒ 1:2 ⇒ 2:2 ⇒ 3:2. Fig. 2 shows the task selection of four robots except robot 1 (*Striker*) and robot 6 (*Goalkeeper*) (Two robots do not change their task because both roles are the highest priority roles in each task).

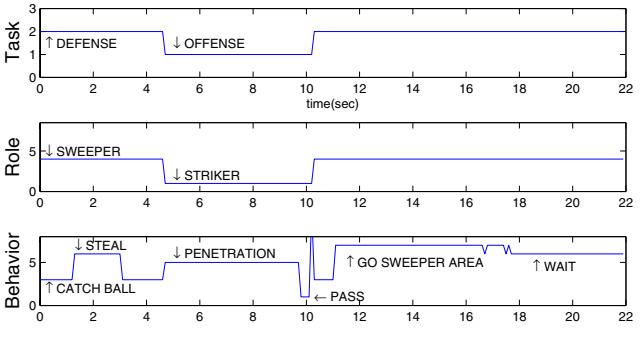
In Fig. 2, the offense task and the defense task were symbolized as 1 and 2, respectively. The robots 2 and 3 were defensive ones and the robot 4 and 5 were offensive ones. The graph shows that the robots 2 and 3 mainly selected defense task and robots 4 and 5 selected offense task. The robots tended to select the

offense task around 15 to 25 seconds because the team was losing the game and the time was running out. This tendency occurred regardless of the strategies. It was because the strategy ratio adjusted weights of offense and defense tasks to adapt the environmental change. And the graph also shows that the team selected defense task more than offense task after 35 seconds because they were winning the game and the time was not left much.

Simulation Game with Two Teams. This simulation shows how robots in a team select a task, a role and a behavior dynamically in adversarial and dynamic environment and how they can help each other when a robot requests a help. In the simulation, there were two teams, BLUE and GREEN, and they both had one offensive robot and two defensive robots. Note that the $robot_i$ in BLUE team and the $robot_j$ in GREEN team were represented as $robot_i^B$ and $robot_j^G$,



(a)



(b)

Fig. 4. Task, Role and Behavior Selections. (a) $robot_1^C$, (a) $robot_2^C$

respectively. And the Fig. 3 is the results of the simulation game which show the task, role and behavior selections of GREEN team robots, $robot_1^G$ and $robot_3^G$.

Experiment with Real Platforms. The omniwheel and omnivision-based robot platform, OmniBot was used for the experiment. It can detect landmarks and other robots by using omni mirror and can dribble a ball by omnidirectional mobility. Two teams (CYAN team and MAGENTA team) were competed with each other and each team consisted of one offensive and one defensive robots. The task, role and behavior selections of CYAN team robots are shown in Fig. 4.

The $robot_i$ in CYAN team and the $robot_j$ in MAGENTA team were represented as $robot_i^C$ and $robot_j^M$ respectively. At first, $Striker_1^M$ had the ball but it was intercepted by $Sweeper_3^C$ and it changed its role to $Striker_3^C$. $Sweeper_3^M$ tried to intercept the ball from $Striker_3^C$ and $Striker_3^C$ passed the ball to $Forward_1^C$. After passing the ball, $Striker_3^C$ changed its role to $Sweeper_3^C$ and $Forward_1^C$ changed its role to $Striker_1^C$ and shot the ball to the goal of the MAGENTA team.

4 Conclusions and Future Works

This paper proposed the multi-robot coordination architecture (MCMRA) for dynamic task, role and behavior selections in adversarial and dynamic environment. The proposed architecture, MCMRA, was applied to robot soccer domain. The task strategy ratio was introduced to show that each robot in a team could select its role adaptively based on the team strategy and the environmental changes. Utilizing the task strategy ratio implied that each robot could consider team work as well as its motivation. The simulation game with two teams showed that they could coordinate each other by considering other robot's motivation, requirement for help and environmental changes such as the opponent team robots and the ball possessing team. The experiment with real robots showed that the proposed architecture could be implementable in real robot team.

For the future work, multiple number of robots should be used in a team to test the robustness and scalability of the proposed architecture. In addition, a general framework for creating motivation and utility by considering the task requirement, cost of the task and robot capability such as quality, quantity and energy consumption of each robot should be studied.

References

1. Parker, L.E.: ALLIANCE: An architecture for fault-tolerant multirobot cooperation. IEEE Trans. on Robotics and Automation 14, 220–240 (1998)
2. Werger, B., Mataric, M.J.: Broadcast of Local Eligibility for Multi-Target Observation. In: Proceedings, 5th International Symposium on Distributed Autonomous Robotic Systems, pp. 347–356 (2000)
3. Gerkey, B.P., Mataric, M.J.: Sold!: Auction Methods for Multirobot Coordination. IEEE Trans. on Robotics and Automation 18, 758–768 (2002)

4. Dias, M.B.: Traderbots: A new paradigm for robust and efficient multirobot coordination in dynamic environments, Ph.D. dissertation, Robotics Institute, Carnegie Mellon University, Pittsburgh (January 2004)
5. Zlot, R.: An Auction-Based Approach to Complex Task Allocation for Multirobot Teams, Ph.D. dissertation, Robotics Institute, Carnegie Mellon University, Pittsburgh (December 2006)
6. Jones, E.G., Browning, B., Bernardine Dias, M., Argall, B., Veloso, M., Stentz, A.: Dynamically formed heterogeneous robot teams performing tightly-coordinated tasks. In: IEEE International Conference on Robotics and Automation, pp. 570–575 (2006)
7. Hecht-Nielsen, R.: Cogent confabulation. Neural Netw. 18, 111–115 (2005)
8. Kim, J.-H., Cho, S.-H., Kim, Y.-H., Park, I.-W.: Two-layered Confabulation Architecture for an Artificial Creature's Behavior Selection. IEEE Trans. on Systems, Man, and Cybernetics - PART C 38(6), 1–7 (2008)
9. Gerkey, B.P., Mataric, M.J.: A formal analysis and taxonomy of task allocation in multirobot systems. International Journal of Robotics Research 23, 939–954 (2004)
10. Kim, J.-H., Kim, D.-H., Kim, Y.-J., Seow, K.-T.: Soccer Robotics (Springer Tracts in Advanced Robotics), p. 326. Springer, Heidelberg (2004)
11. Kim, J.-H., Lee, K.H., Kim, Y.D.: The Origin of Artificial Species: Genetic Robot. International Journal of Control, Automation, and Systems 3(4), 564–570 (2005)

A Robot Localization Method Based on Laser Scan Matching

David Sandberg, Krister Wolff, and Mattias Wahde

Department of Applied Mechanics
Chalmers University of Technology, 412 96 Göteborg, Sweden
{david.sandberg,krister.wolff,mattias.wahde}@chalmers.se

Abstract. The problem of robot localization has been considered for applications (such as, for example, transportation and delivery) where the robot is given, *a priori*, a map of the arena in which it is operating. A localization method based on laser scan matching is described, in which localization is carried out using a simple stochastic search algorithm. The results from multiple experiments show that the proposed method can correct position errors of up to 0.7 m and, more importantly, also can correct errors in heading angle, thus achieving negligible heading errors in most cases.

Keywords: Robot localization, laser scan matching, robot pose estimation.

1 Introduction and Background

A mobile robot operating in an arena (known or unknown) must normally keep track of its position and heading[1]. In some cases, the arena is initially unknown, and one of the robot's first tasks can then be to explore the arena while keeping track of its position; in other words, the robot must execute *simultaneous localization and mapping* (SLAM) [4]. In other situations (e.g. the one considered in this paper), a map may be provided to the robot *a priori*, and the robot's task is then to move in the arena (executing, for instance, a transportation task) while carrying out *localization*, i.e. maintaining an accurate pose estimate.

In mobile robot navigation, it is common to use odometry, that is, determining the robot's pose using integration of the kinematic equations. In turn, odometry requires measurements of the rotation of the wheels, as obtained by wheel encoders mounted on each motor axis. As is well known, odometry is never perfectly accurate. Sources of error include wheel slip and the fact that the wheel encoders have limited resolution. Furthermore, the kinematic model itself may not be perfect. Thus, the odometric estimates will suffer from an ever-growing error, a phenomenon known as *odometric drift*.

[1] The position and heading together form the *pose* of a robot. Note that, in this paper, we shall only consider differentially steered robots, with two powered wheels and one or two support points.

J.-H. Kim et al. (Eds.): FIRA 2009, LNCS 5744, pp. 171–178, 2009.

Therefore, the robot must, from time to time, correct its odometric pose estimate. Assuming that a map is available (either by an earlier execution of SLAM, or simply provided by the user), the correction can be carried out by means of *scan matching*, i.e. a procedure that matches the readings of a (two-dimensional) laser range finder (LRF) to the map, which is typically represented as a set of lines defining a horizontal slice through the arena.

In deterministic methods for localization, a single estimate of the robot's pose is maintained (rather than the probabilistic density functions used, for example, in probabilistic SLAM methods [4]). Such methods have been considered by, among others, Lu and Milios [3] and Cox [2]. Many of these methods (e.g. the one described by Lu and Milios [3]) are derived from (or at least inspired by) the *iterative closest point* (ICP) algorithm, originally developed in the field of computer vision (see [1] for a description of ICP). In deterministic methods, a measure of the pose error is defined based on the discrepancy between the actual sensor readings and the map. Once the error measure has been defined, the aim is to find a pose that minimizes the error. This pose is then taken as the true pose of the robot, allowing a correction of the odometry.

Regarding the procedure for minimizing the pose error, several different approaches have been used in the literature. For example, Lu and Milios [3] used least-squares linear regression whereas Ze-Su *et al.* [6] used stochastic optimization (in the form of a genetic algorithm). However, setting up and running a full-fledged population-based stochastic optimization method (such as a genetic algorithm or particle swarm optimization) may be too time-consuming, especially in cases (such as the one described below) where the map is rather complex. In this paper, we shall introduce a simple stochastic search algorithm for the problem of finding the actual pose of a robot (*given* a map of the arena), using scan matching.

2 Problem Formulation

Consider, for example, a robot involved in a transportation task, delivering objects from an arbitrary starting point (A) in an arena to a given target point (B). The robot must, of course, not only be able to chart a course between A and B, but must also keep track of its position and heading along the way. In the remainder of this paper, we shall consider cases in which a robot is provided with a map (in the form of a sequence of lines), corresponding to a horizontal *slice* through the arena, at the height of the LRF that is assumed to be mounted on top of the robot. No restrictions are introduced regarding the manner in which the map has been generated: It can either have been generated by the robot itself *or* by the user, for example on the basis of the floor plan for the building, or part thereof, in which the robot is operating[2]. Returning to the transportation task, the robot would navigate toward the target (point B) using, say, A*-navigation based on odometric estimates, which are assumed to be available at all times (i.e. the robot is assumed to be equipped with wheel encoders).

[2] Note that only planar, indoor environments are considered in this paper.

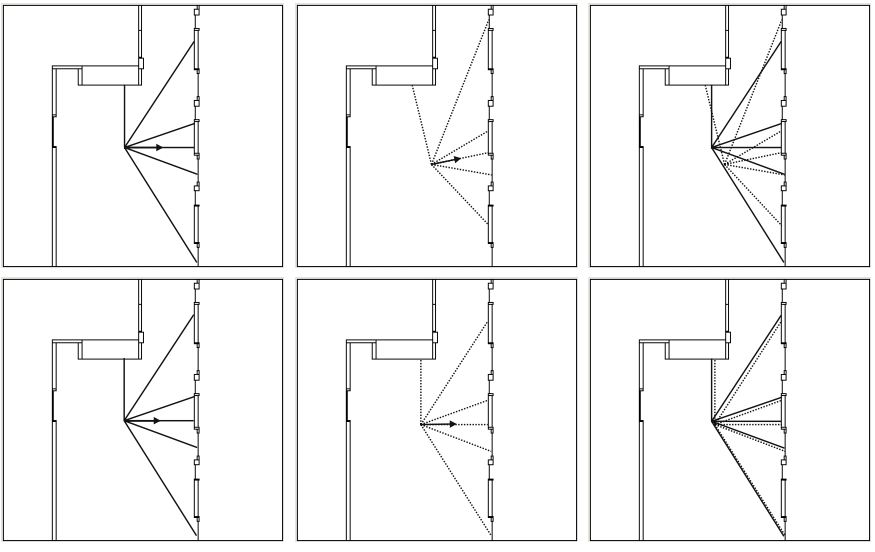


Fig. 1. Two examples of scan matching. The leftmost panel in each row shows a few rays (solid lines) from an actual LRF reading, and the middle panels show the virtual LRF readings (dotted lines) in a case in which the estimated pose differs quite strongly from the correct one (upper row), and one in which the difference is small (bottom row). The direction of heading is illustrated with arrows. The right panel in each row shows both the actual LRF rays and the virtual ones. The figure also illustrates the map, which consists of a sequence of lines.

However, because of the inevitable odometric drift, the robot must periodically activate a localization behavior, in order to correct the odometric readings. When activated, the localization behavior considered here first stops the robot, and then takes a reading of the LRF. Next, it tries to match this reading to a *virtual* reading taken by placing a virtual LRF (hereafter: vLRF) at various positions *in the map*[3]. Two examples of scan matching are shown in Fig. 1. The three panels in the upper row show a situation in which the odometry has drifted significantly. The upper left panel shows the readings (i.e. laser ray distances) from an actual LRF mounted on top of the robot (not shown). Note that, for clarity, the figure only shows a few of the many hundreds of laser ray directions (see also Fig. 2). The upper middle panel shows the readings of the vLRF, placed at the initial position and heading obtained from odometry. As can be seen in the upper right panel, the two scans match rather badly. By contrast, the three panels of the bottom row show a situation in which the pose error is small. The purpose of the search algorithm described below is to be able to correct the odometry, i.e. to reach a situation similar to the one shown in the bottom row of Fig. 1.

[3] Obviously, this behavior requires that a rather accurate map is available and that the sensor can be modelled accurately (for the virtual scans). Fortunately, the error of a typical LRF is sufficiently small to make the procedure outlined in this paper feasible.

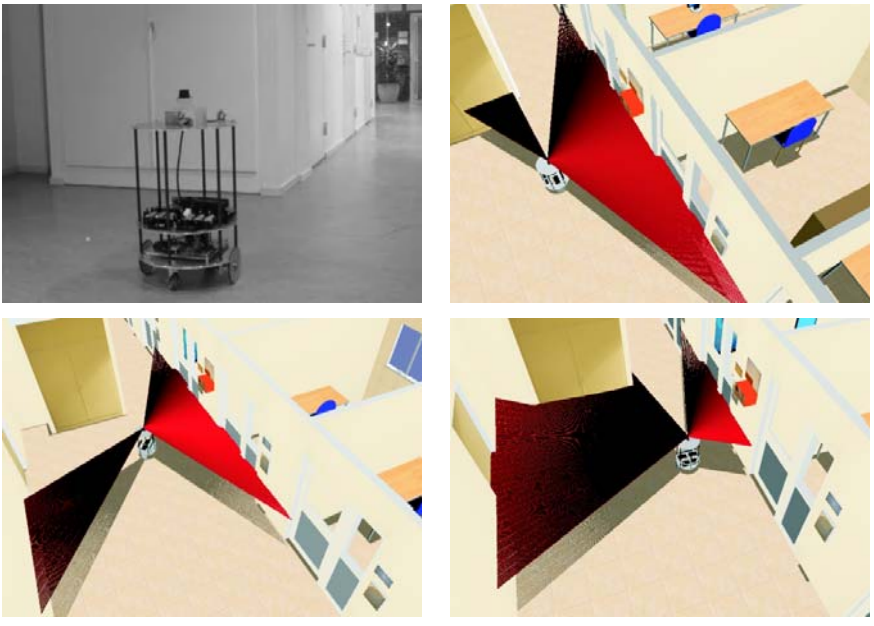


Fig. 2. Upper left panel: The robot used in the experiments. The three remaining panels show the pose of the robot in the three experiments. Upper right panel: Experiment 1; lower left panel: Experiment 2; lower right panel: Experiment 3. Note that *all* measurements used in the experiments were taken from the real Hokuyo LRF, not from the simulator.

In earlier work [5], we have proposed and tested (using a Hokuyo URG-04LX LRF mounted on a Pioneer robot base) a brute-force search method, in which the first virtual scan is taken at the pose corresponding to the current estimated position (from odometry, before correction), and where subsequent scans are carried out in an expanding circular pattern around the first scan. While successful, such a procedure is not very efficient, since it does not, in any way, make use of the improvements (i.e. reductions in the matching error) that might be found during the search. In this paper, a more efficient search procedure will be introduced.

3 Method

Let $\mathbf{p} = (x, y, \varphi)$ denote a pose (in the map) of the vLRF. The distances between the vLRF and an obstacle, along ray i, are obtained using the map[4] and are

[4] In practice, the ray reading δ_i of the vLRF is obtained by checking for intersection between the lines in the map and a line of length R (the range of the LRF) pointing in the direction of the ray, and then choosing the shortest distance thus obtained (corresponding to the nearest obstacle along the ray). If no intersection is found, the corresponding reading is set to -1.

denoted δ_i. Similarly, the distances obtained for the *real* LRF (at its current pose, which normally differs from $\mathbf{p}$ when the localization behavior is activated) are denoted d_i.

The matching error ϵ between two scans can be defined in various ways. For rays that do not intersect an obstacle, the corresponding reading (d_i or δ_i) is (arbitrarily) set to -1. Such rays should be excluded when computing the error. Thus, the matching error is taken as

$$\epsilon = \sqrt{\frac{1}{\nu}\sum_{i=1}^{n}\chi_i\left(1-\frac{\delta_i}{d_i}\right)^2}, \tag{1}$$

where n is the number of LRF rays used[5]. The parameter χ_i is equal to one for those indices i for which *both* the physical LRF and the vLRF detect an obstacle (i.e. obtain a reading different from -1) whereas χ_i is equal to zero for indices i such that either the real LRF or the vLRF (or both) do not detect any obstacle (out to the range R of the LRF). ν denotes the number of rays actually used in forming the error measure, i.e. the number of rays for which χ_i is equal to one. As can be seen, ϵ is a measure of the (normalized) average relative deviation in detected distances between the physical LRF and the vLRF.

Since d_i are given and δ_i depend on the pose of the vLRF, one may write $\epsilon = \epsilon(\mathbf{p})$. Now, if the odometric pose estimate happens to be exact, the virtual and actual LRF scans will be (almost) identical (depending on the accuracy of the map), giving a very small matching error, in which case the localization behavior can be deactivated and the robot may continue its navigation. However, if the error exceeds a user-defined threshold T, the robot can conclude that its odometric estimates are not sufficiently accurate, and it must therefore try to minimize the matching error by trying various poses in the map, i.e. by carrying out a number of virtual scans, in order to determine the *actual* pose of the robot. The scan matching problem can thus be formulated as the optimization problem of finding the pose $\mathbf{p} = \mathbf{p}_v$ that minimizes $\epsilon = \epsilon(\mathbf{p})$. Once this pose has been found, the new odometric pose $\mathbf{p}^{\text{new}}$ is set equal to $\mathbf{p}_v$.

Note that it is assumed that the robot is standing still during localization. This restriction (which, in principle, can be removed) is introduced in order to avoid having to correct for the motion that would otherwise take place during scan matching. Thus, only *one* scan needs to be carried out using the real LRF mounted on the robot. The remaining work consists of generating virtual scans in the map, at a sequence of poses, and to match these to the actual LRF readings. Unlike some other scan matching methods (see e.g. [2,3]), the method used here does not attempt to fit lines to the LRF readings. Instead, the LRF rays (actual and virtual, as described above) are used directly during scan matching. The sequence of poses for the vLRF is generated using the algorithm described next.

[5] For example, in the case of a Hokuyo URG-04LX LRF, a maximum of 682 rays are available.

3.1 Algorithm

Here, a very simple algorithm is used: First the actual LRF scan is carried out, generating the distances d_i. Next, a virtual scan is carried out (in the map) at the current estimated position $\mathbf{p}_0$. If the error $\epsilon_0 = \epsilon(\mathbf{p}_0)$ is below the threshold T, localization is complete. If not, the algorithm picks a random pose $\mathbf{p}_j$ (where $j = 1$ in the first iteration) in a rectangular box of size $L_x \times L_y \times L_\varphi$, centered on $\mathbf{p}_0$ in pose space, and computes the matching error $\epsilon_j = \epsilon(\mathbf{p}_j)$. This is repeated until, for some $j = j_1$, an error $\epsilon_{j_1} < \epsilon_0$ is found. At this point, the rectangular box is re-centered to $\mathbf{p}_{j_1}$, and the search continues, now picking a random pose in the rectangular box centered on $\mathbf{p}_{j_1}$. Once a position $\mathbf{p}_{j_2}$ is found for which $\epsilon_{j_2} < \epsilon_{j_1}$, the rectangular box is again re-centered etc. The procedure is repeated for a given number (N) of iterations[6].

4 Results and Discussion

The localization method was tested by placing the robot in three different poses in a given arena, and then running the search algorithm. The runs were carried out using a computer program that would first open a stored *actual* LRF reading as well as the map (see below) then carrying out a number of executions of the search algorithm, starting from different initial pose estimates, corresponding to different initial odometric errors.

A detailed map was generated in the form of a sequence of lines slicing the arena horizontally at the height of the LRF mounted on the robot. During execution of the program, the virtual scans (obtained for the vLRF poses generated by the search algorithm) are matched to the real scan for the experiment in question. Fig. 2 shows the actual poses of the robot in the three experiments. Note that all three measurements were taken using a real Hokuyo LRF: The simulator screenshots in Fig. 2 are used only for the purpose of visualizing the LRF rays which, of course, are invisible in the case of a real LRF. The results obtained are summarized in Table 1. In all runs, the algorithm was allowed to run for $N = 60$ iterations, corresponding to about 2 s of execution time on a laptop with a 2.4 GHz Core 2 Duo processor. Note that the experiments were carried out using rather large initial pose errors, in order to make sure that the algorithm could handle such situations. Some examples involving smaller initial error can be seen in Fig. 3 below.

The table shows that, with a few rare exceptions (occurring at large initial *heading* error), the algorithm manages to reduce the position error, sometimes to 0.10 m or less. In addition, again with a few rare exceptions, the algorithm strongly reduces the heading error. Note that the strong correction in heading angle is especially important, since an incorrect heading quickly leads to an increasing position error as the robot starts to move. Several different values of the parameters L_x, L_y and L_φ (see Sect. 3.1) were tested. The runs shown in the

[6] Note that this algorithm resembles an evolutionary algorithm with a population size of one.

Table 1. Columns 1 and 2 show the initial odometric pose error (radial position error and heading angle error), and the remaining columns show the final pose error obtained in Experiments 1, 2 and 3. Each entry in columns 3-8 corresponds to an average of 80 runs (16 different odometric positions with an initial radial error of e_r^0, and 5 runs for each position). The results are given as averages, listed with a 95% confidence interval (assuming a normal distribution).

		Exp. 1		Exp. 2		Exp. 3	
e_r^0	e_φ^0	e_r^f	e_φ^f	e_r^f	e_φ^f	e_r^f	e_φ^f
0.3	-0.785	0.29 ± 0.05	0.35 ± 0.10	0.26 ± 0.04	0.02 ± 0.01	0.15 ± 0.03	0.03 ± 0.01
0.3	-0.393	0.07 ± 0.01	0.01 ± 0.00	0.23 ± 0.03	0.01 ± 0.00	0.09 ± 0.01	0.02 ± 0.00
0.3	0.000	0.07 ± 0.01	0.01 ± 0.00	0.23 ± 0.02	0.01 ± 0.00	0.08 ± 0.01	0.01 ± 0.00
0.3	0.393	0.07 ± 0.01	0.02 ± 0.00	0.24 ± 0.03	0.01 ± 0.00	0.10 ± 0.01	0.01 ± 0.00
0.3	0.785	0.41 ± 0.07	0.75 ± 0.18	0.30 ± 0.04	0.05 ± 0.05	0.20 ± 0.03	0.04 ± 0.01
0.5	-0.785	0.29 ± 0.05	0.25 ± 0.08	0.32 ± 0.04	0.02 ± 0.00	0.20 ± 0.03	0.03 ± 0.01
0.5	-0.393	0.11 ± 0.02	0.02 ± 0.00	0.36 ± 0.04	0.01 ± 0.00	0.15 ± 0.03	0.02 ± 0.00
0.5	0.000	0.07 ± 0.01	0.01 ± 0.00	0.31 ± 0.04	0.01 ± 0.00	0.11 ± 0.01	0.02 ± 0.00
0.5	0.393	0.10 ± 0.01	0.02 ± 0.00	0.32 ± 0.04	0.01 ± 0.00	0.14 ± 0.02	0.02 ± 0.00
0.5	0.785	0.40 ± 0.07	0.80 ± 0.18	0.44 ± 0.06	0.16 ± 0.10	0.21 ± 0.04	0.03 ± 0.01
0.7	-0.785	0.42 ± 0.07	0.36 ± 0.09	0.51 ± 0.07	0.08 ± 0.06	0.30 ± 0.05	0.06 ± 0.01
0.7	-0.393	0.14 ± 0.02	0.02 ± 0.00	0.47 ± 0.05	0.01 ± 0.00	0.19 ± 0.03	0.03 ± 0.00
0.7	0.000	0.11 ± 0.02	0.02 ± 0.00	0.45 ± 0.05	0.01 ± 0.00	0.17 ± 0.03	0.03 ± 0.00
0.7	0.393	0.14 ± 0.02	0.02 ± 0.00	0.48 ± 0.05	0.01 ± 0.00	0.20 ± 0.03	0.03 ± 0.01
0.7	0.785	0.46 ± 0.07	0.80 ± 0.83	0.51 ± 0.05	0.29 ± 0.14	0.30 ± 0.04	0.05 ± 0.01

table were carried out with the values that turned out to give the best results, namely $L_x = L_y = 0.10$ (m) and $L_\varphi = 0.10$ (radians). An attempt was also made to improve the results by letting the algorithm run for an additional 30 iterations (i.e. a total of 90). The results were mixed, showing a small improvement in the case of small initial errors in heading angle but no improvement for larger ($\pm\pi/4$) initial errors in heading angle.

Fig. 3 shows the detailed sequence of intermediate points obtained for two of the many thousands of runs carried out. The correct position is shown as a filled square, with the arrow indicating the heading angle. The initial estimated pose is indicated by the filled disc (and its accompanying arrow), and the intermediate poses are indicated using open discs. The left panel shows a case in which the initial position error was $e_r^0 = 0.20$ (m), and the initial heading error was $e_\varphi^0 = \pi/12$. In the right panel the position error was $e_r^0 = 0.10$ (m) and the heading error was $e_\varphi^0 = \pi/6$. As can be seen, the algorithm takes the estimated pose on a rather wide-ranging excursion, while correcting the estimated heading angle, before converging (after 60 iterations, in both cases) to the final estimate, shown as an open square. The radial position errors were thus reduced to 0.030 m (left panel) and 0.038 m (right panel), whereas the heading errors were reduced to 0.007 radians (left panel) and 0.021 radians (right panel).

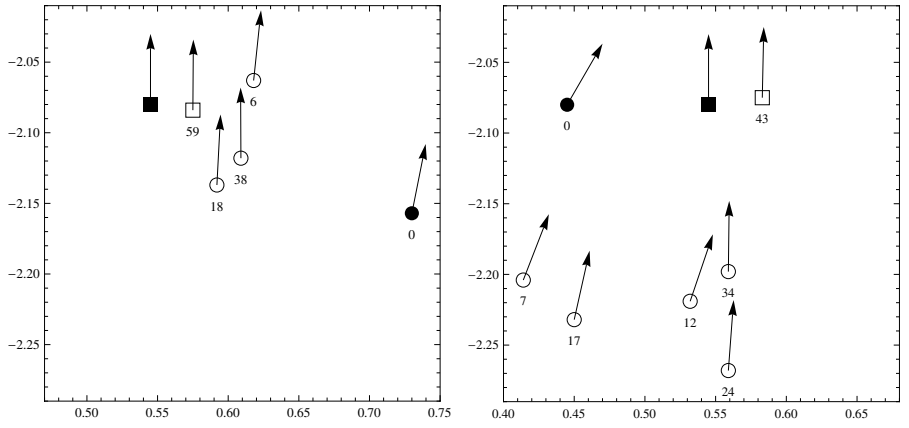


Fig. 3. An illustration of the sequence of poses generated during two executions of the search algorithm (with arrows indicating the direction of heading). In each panel, the actual position (measured in meters) of the robot is indicated with a filled square. The initial estimated position (i.e. from odometry, before correction) is shown as a filled disc, and the final estimated position is visualized as an open square. The intermediate points generated during the search are represented as open discs and are shown together with the corresponding iteration number. Note that, for clarity, only some of the intermediate points are shown.

In conclusion, one may note that the simple search algorithm used here is able to correct odometric pose errors reliably and relatively quickly. Most importantly, the algorithm strongly reduces the error in heading angle, in the vast majority of cases.

References

1. Besl, P.J., McKay, N.D.: A Method for Registration of 3-D Shapes. IEEE Trans. on Pattern Analysis and Machine Intelligence 14(2), 239–256 (1992)
2. Cox, I.J.: Blanche – An Experiment in Guidance and Navigation of an Autonomous Robot Vehicle. IEEE Trans. on Robotics and Automation 7(2), 193–204 (1991)
3. Lu, F., Milios, E.J.: Robot Pose Estimation in Unknown Environments by Matching 2D Range Scans. Intell. Robotic. Syst. 18(3), 249–275 (1997)
4. Thrun, S., Burgard, W., Fox, D.: Probabilistic Robotics. MIT Press, Cambridge (2005)
5. Pettersson, J., Hartono, P., Wahde, M.: A Behavior Module for Odometry Recalibration in Autonomous Robots. In: Proc. of AMiRE 2007, pp. 11–17 (2007)
6. Ze-Su, C., Bing-Rong, H., Hong, L.: An Improved Polar Scan Matching Using Genetic Algorithm. Inform. Tech. J. 6(1), 89–95 (2007)

An Algorithm for Sensory Area Coverage by Mobile Robots Operating in Complex Arenas

Mattias Wahde and David Sandberg

Department of Applied Mechanics
Chalmers University of Technology, 412 96 Göteborg, Sweden
{mattias.wahde,david.sandberg}@chalmers.se

Abstract. In this paper, a novel algorithm for sensory area coverage by mobile robots is proposed, with applications in, for example, mapping and exploration. The algorithm generates a sequence of nodes for the robot to visit, such that it covers as much as possible of the arena using a laser range finder. A crucial part of the exploration behavior implementing this algorithm is a deadlock avoidance procedure, which allows the robot to handle the inevitable problems (such as coping with narrow passages and obstacles of different size and height) that occur when navigating in complex arenas. Despite the simplicity of the algorithm, the robot is generally able to cover (with its laser range finder) 98% or more of an initially unexplored arena.

Keywords: Robot exploration algorithms, area coverage.

1 Introduction and Background

The problem of *area coverage* appears in many contexts involving mobile robots such as, for example, lawn mowing, vacuum cleaning, clearing mine fields etc. In such applications, the robot is required to physically cover the entire arena. Methods for physical area coverage can be divided into grid-based methods, which rely on an occupancy grid covering the arena (see e.g. [5]), and topological methods that rely on landmarks (see e.g. [1]).

However, for the purposes of mapping, exploring, or guarding an arena, it is unnecessary for the robot to physically visit every spot in the arena. Instead, what matters is that its long-range sensor(s), for example a laser range finder (LRF), is able to reach every place in the arena at some point during the robot's motion (for mapping and exploration) or periodically (for guarding applications where, for example, a robot is used as a night watchman). An important special case is the problem of simultaneous localization and mapping (SLAM) where a robot is required to build a map of an initially unknown arena, while keeping track of its pose[1]. However, as noted by González-Baños and Latombe [3], SLAM methods generally do not specify exactly *how* the robot should move when exploring the arena.

[1] The position and heading together form the *pose* of a robot.

J.-H. Kim et al. (Eds.): FIRA 2009, LNCS 5744, pp. 179–186, 2009.

Thus, the problem of exploring an arena such that the long-range sensor(s) reach all points in an arena, which here will be called the *sensory area coverage* (SAC) problem[2], is relevant both in SLAM and in applications (such as guarding) where a map might already be available.

The SAC problem has been considered by several authors and, as in physical area coverage, the methods used are typically either grid-based [4] or topological [2]. For a brief review of SAC, see [4]. The literature on SAC is mostly concerned with the mathematical properties (such as guaranteed convergence, time-to-completion etc.) of the algorithms considered. However, when applied to a real robot operating in a complex arena (with, for example, furniture and other obstacles in addition to the walls defining the arena), several practical problems occur. For example, a 2D LRF (which is a commonly used sensor in connection with SAC) will, in effect, only see a two-dimensional (horizontal) slice through the arena. Thus, even though the 2D LRF may indicate that a particular path is passable, there might still be obstacles whose height is smaller than that at which the LRF is mounted. In order to avoid collisions with such obstacles, the robot must be equipped with proximity detectors at various heights. In addition, however, the SAC algorithm used must be able to cope with situations in which a path must be abandoned due to low-lying obstacles. Furthermore, even though a particular node (i.e. a waypoint) on a path does not trigger a proximity detection when approached from one direction, it may do so if the robot approaches the node from a slightly different direction, or if the robot takes a slight shortcut; in a realistic SAC algorithm one cannot assume that the robot reaches each node along its path *exactly*. Instead, a node is considered to be reached when the robot passes it within some threshold distance.

The problem of generating a robust SAC algorithm is of interest in its own right and can be studied separately from the problem of localization (just like, as mentioned above, SLAM methods generally do not specify the exploration algorithm used!) Thus, in order to isolate the SAC problem, in this paper we shall assume that the odometric readings of the robot do not drift indefinitely (a situation that can be achieved, for example, using a Northstar[3] localization system).

2 Mobile Robot Simulator

The robot that will be considered here is a differentially steered (wheeled) mobile robot equipped with a (two-dimensional) LRF, and a set of IR sensors for proximity detection. Such a robot, shown in the left panel of Fig. 1, is currently under construction in the authors' research group. However, in view of the many runs needed for testing the proposed SAC behavior, simulations were used rather than the real robot, for the work presented here. Another reason for using simulations is that reconfiguring a robot is, of course, simpler in a simulator than

[2] In the literature, this problem has sometimes been referred to as the *next-best-view* problem [3].

[3] See `www.evolution.com/products/northstar/`

Fig. 1. Left panel: The mobile robot currently under construction in the authors' research group. Right panel: A screenshot from GPRSim, showing the arena used in the simulations. The robot can be seen in the upper right corner.

in a real robot; an additional aim of the simulations presented here is thus to identify appropriate IR proximity sensors, and to place them in the optimal positions (in the simulated robot), before actually attaching them to the real robot. The simulations were carried out using the *General-purpose robotic simulator* (GPRSim) also developed in the authors' research group [6]. A screenshot from the simulator can be seen in the right panel of Fig. 1. GPRSim is able to simulate accurately various differentially steered robots equipped with a variety of sensors and actuators, with appropriate noise levels (obtained through system identification of the corresponding physical components).

3 Algorithm

The proposed algorithm operates as follows: Upon initialization, the robot generates a set of nodes, at a distance D from its current estimated position (as obtained using odometry). A maximum of N nodes are distributed, at equidistant angles over an angular range α, which should be no larger than the scan angle of the LRF (typically around 180-240 degrees, depending on the specific type of LRF used). However, nodes are only placed in those directions in which the distance to the nearest obstacle (as obtained from the LRF) exceeds $D + R$, where R is the (effective) *robot size* (i.e. the robot's radius plus a margin). The node at which the robot is currently located is given the status *completed* (meaning that it has been reached) and is referred to as the *active node*. All newly added nodes are given the status *pending*. The robot also generates paths to the pending nodes. For example, if the robot is located at node 1 and generates three pending nodes (2, 3, and 4), the paths will be (1,2), (1,3), and (1,4). The robot next selects the nearest node, based on the *path length*, as the current *target node*[4]. Next it rotates (if needed) to face the target node, and then proceeds

[4] At this stage, the three pending nodes available are all at the same distance, in which case the robot randomly selects one of them as the target node.

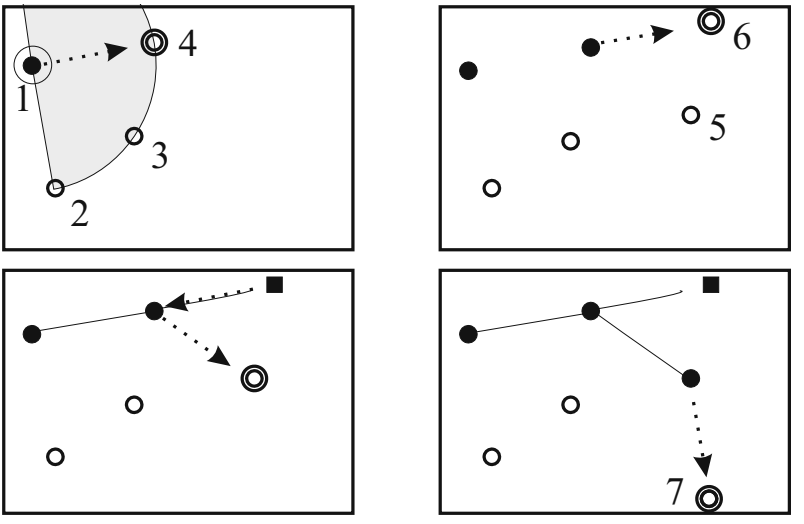


Fig. 2. The early stages of a run using the proposed SAC algorithm, showing a robot exploring a single rectangular room. In each panel, the target node is shown with two thick circles, pending nodes are shown as a single thick circle, completed nodes as a filled disc, and unreachable nodes as a filled square. Upper left panel: The robot, whose size is indicated by a thin open circle in this panel, starts at node 1, generating three new pending nodes (2,3 and 4). The scan sector in which nodes can be generated is indicated in this panel, to highlight the fact that nodes are only generated in those directions in which the robot has a clear view out to a distance of at least $D+R$. Upper right panel: Having reached node 4, the robot sets the status of that node to *completed*, and then generates new pending nodes. Lower left panel: Here, the robot has concluded (based on IR proximity readings) that node 6 is unreachable, and it therefore selects the nearest pending node (5, in this case) as the new target node. Lower right panel: Having reached node 5, the robot can only generate one new node (7). It rotates to face that node, and then moves toward it etc.

towards it; see the upper left panel of Fig. 2. During this motion, one of two things can happen[5]: Either (i) the robot reaches the target node or, (ii) using its IR proximity detectors, it concludes that the target node cannot be reached along the given path.

In case (i), illustrated in the upper right panel of Fig. 2, once the target node has been reached, it is given the status *completed* and is then set as the new active node. At this point, the paths to the remaining pending nodes are updated. Continuing the example above, if the robot moves to node 4, the paths to the pending nodes 2 and 3 will be updated to (4,1,2) and (4,1,3). Furthermore,

[5] It is assumed that the arena remains fixed and does not contain any moving obstacles during the exploration phase.

the robot now generates new pending nodes, using node 4 as the center point[6], again only placing nodes in those directions in which the LRF has a clear view out to a distance of at least $D + R$ from the current pose. Proceeding with the example, the robot may at this stage generate nodes 5 and 6, with the paths (4,5) and (4,6). Again, the robot selects the nearest node based on the path length[7], rotates to face that node, and proceeds toward it etc. Note that the robot may (and will) visit completed nodes more than once. However, by the construction just described, only pending nodes can be target nodes.

In case (ii), where a node cannot be reached, it is assigned the status *unreachable*, and the robot instead assigns a new target node and proceeds toward it along the path. This situation is illustrated in the upper right and lower left panels of Fig. 2: Here, the robot concludes that it cannot reach node 6. It thus marks this node as unreachable, sets it as the active node, and then sets the nearest (based on path length) pending node as the new target, in this case node 5. The reader may wonder how case (ii) can occur, since the robot uses the LRF scan (as described above) before assigning nodes. However, if the robot's laser scan hits a wall at an oblique angle (as in the placement of node 6 in Fig. 2), a node may be placed too near a wall, even though the scan distance exceeds $D + R$. Furthermore, the two-dimensional LRF only scans the arena at a given height. A low obstacle may therefore be missed, until the IR proximity detectors (mounted at different heights) perceive it. In addition to the restrictions on node placements already considered, a minimal cartesian distance d between any pair of (pending or completed) nodes is enforced, making the algorithm complete its task in finite time: Every time a *target* node is reached, the robot attempts to generate new nodes as described above, but if the distribution of nodes (pending or completed) already present is such that any new node (placed according to the method described above) would be closer than d to the active node, no node is added. However, note that unreachable nodes are excluded from the minimal distance requirement. This is so, since a location may be unreachable from one direction, but perhaps reachable from another. Thus, the algorithm is allowed to place new (pending) nodes arbitrarily close to unreachable nodes.

Because of the path following navigation strategy, in some cases the robot may take an unnecessarily long path from the active node to the target node, but this happens less often than one might think: In most cases, the robot will proceed to a newly generated node, for which the path length is the same as the cartesian distance. However, whenever the robot is unable to place new nodes (having reached, say, a corner), a far-away node often becomes the target node. Therefore, in such cases, where the path to the target node differs from the direct path, the robot first rotates to face the target node (provided that it is located within a distance L, where L should be smaller than the range of the

[6] It is not necessary (or realistic) to demand that the robot should be located *exactly* at the node. In general, a node is considered reached if (according to the odometric estimate) the robot passes it within a distance a.

[7] Selecting (randomly) one of the equidistant nodes 5 and 6, rather than nodes 2 or 3 for which the *path lengths* (from node 4) are longer.

LRF). If, using the LRF, it estimates that the target node can be reached, the robot may then proceed directly towards it, instead of following the path. Also in this case, however, it is possible that a (low) obstacle prevents the robot from reaching the target, in which case the robot instead switches to following the path as described above.

The algorithm just described was implemented in GPRSim in the form of a finite-state machine containing 17 states (which will not be described in detail here). In addition to the algorithm just introduced, the SAC behavior implemented in GPRSim also includes a procedure for handling deadlocks that might occur, for example, in cases where the robot drifts slightly off its course (remembering that the robot never passes *exactly* over nodes, but instead at a small distance from them), making a previously passable path segment impassable, as indicated by the robot's IR proximity detectors. In such cases, the robot makes several attempts to solve the problem, simply by slightly backing up in a randomly selected direction, then again attempting to follow the path. This simple procedure is generally sufficient to overcome the deadlock. However, there is also a failure state, which becomes active in the (rare) cases in which the robot finds itself stuck (i.e. where not even the deadlock avoidance procedure succeeds). In such cases, the robot reports its failure, and then stops.

4 Results

The SAC behavior was tested in a large set of simulation runs, using a realistic arena representing an office environment, with desks, chairs, bookshelves etc., but no moving obstacles (e.g. people). As mentioned above, the wheel encoders from which odometric estimates are obtained were taken to be noise-free, in order to isolate the problem of exploration (considered here) from the problem of localization. However, in all other sensors (i.e. the LRF and the IR proximity sensors), realistic noise levels were used, and the proximity detection was based on a moving average of IR readings.

The tunable parameters of the proposed SAC algorithm, introduced in Sect. 3 above, are the distance from the active node to the new pending nodes (D), the effective robot size (R) (i.e. the radius plus a margin), the maximum number of new nodes generated at a target node (N), the angle over which the new nodes are distibuted (α), the minimum internode distance (d), the maximum distance for direct movement between the active node and a target node (L), and the distance (estimated from odometry) at which a node is considered reached (a). In the experiments presented here, the range of the LRF was set to 4 m, and its scan angle to 180 degrees[8].

Appropriate values of the tunable parameters were found through a process of trial-and-error, starting the robot at a given pose in the arena. In order to measure the robot's performance, a fine-meshed virtual grid was placed over

[8] Many LRFs have a larger scan angle, but some do not; thus the opening angle was taken as 180 degrees in order to make sure that the behavior will work with most LRFs.

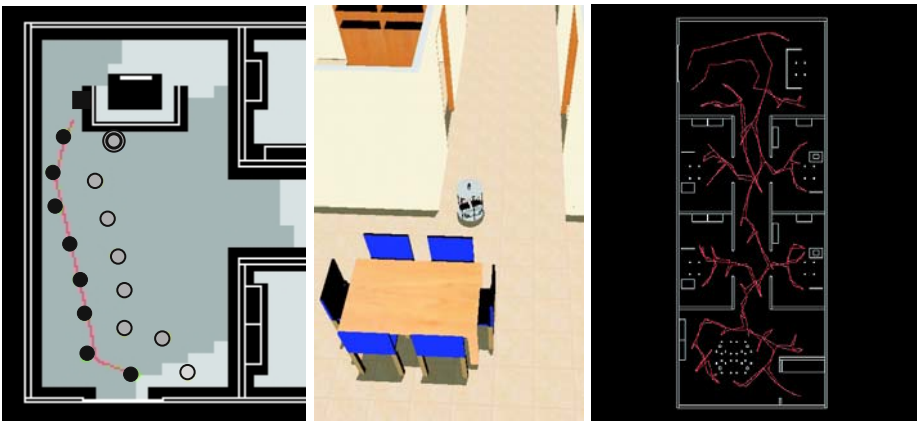


Fig. 3. The left panel shows the early stages of Run 1. The light gray areas indicate parts of the arena that have yet to be covered by the LRF, whereas the dark gray areas indicate parts that have already been covered. Pending nodes are shown as gray discs, and completed nodes are shown as black discs. The target node is shown as a double circle, and unreachable nodes (of which there is only one in this figure) are shown as squares. The middle panel shows the robot in action at a later stage in the same run, as it just reaches one of the most difficult parts of the arena with many narrow obstacles in the form of chairs, a table etc. The entire path of the robot is shown in the right panel.

the arena[9], and a performance measure (c) was then defined as the fraction of (reachable)[10] cells covered (at some point during exploration) by the *robot's range finder*. In addition, two other performance measures were used, namely the time T needed to complete the exploration of the arena and the total length P of the path followed during exploration.

The result for the best parameter settings found (based on the coverage measure c) is given on the first row of Table 1. With these parameter settings, the robot managed to cover 99.5 % of the arena with its LRF in roughly 13 minutes, having travelled a total distance of around 149 m. Many other parameter settings were tried as well, a few of which are shown in the table. In many cases, the robot failed to enter one (or, in rare cases, two) of the rooms, but nevertheless generally managed to achieve 98% (or better) *sensory* area coverage. For example, in Run 4, even though the robot avoided entering one of the rooms in the arena, it reached the door opening of that room, thus covering must of it with its LRF. In rare cases, the robot failed altogether, for example by getting stuck in a narrow passage between two obstacles. The path from Run 1 can be

[9] Note that the robot was not given any information regarding this grid: The SAC algorithm defined here is *not* grid-based.

[10] Only cells that could theoretically be reached were used in the performance measure: Cells that were located, say, inside a piece of furniture or a wall, were of course excluded.

Table 1. The first row shows the best parameter settings found, and the remaining rows give examples of the performance obtained with other parameter settings. Note that the sensory coverage exceeds 98% in all cases shown.

Run	D	R	N	α	d	L	a	c	T	P	Remark
1	1.00	0.90	15	π	0.75	2.50	0.25	0.995	798.56	148.69	Entered all rooms
2	1.25	1.00	21	π	1.00	2.50	0.25	0.981	531.78	110.05	Failed to enter one room
3	1.00	0.90	13	π	0.70	2.50	0.20	0.984	771.00	147.02	Failed to enter one room
4	1.00	0.90	15	π	0.70	2.50	0.20	0.992	828.00	157.09	Failed to enter one room

seen in the right panel of Fig. 3. In all runs, the robot was required to use its deadlock avoidance procedure at least once, proving its importance in realistic exploration tasks.

5 Discussion and Conclusion

In the runs presented here, a single arena (albeit a rather difficult one) was used. In order to ascertain that the parameters presented in Table 1 have general validity, the robot should be tested in several different arenas, a process that is currently underway and will involve stochastic optimization. An obvious improvement, also currently under consideration, is to use a more dynamic node-placement strategy, using varying values of D, N etc. depending on the sensor readings obtained at the current pose, for example making use of landmarks such as doorways when placing nodes. However, one should also keep in mind that, in real robots operating in realistic environments (such as the arena used here), increases in complexity are often accompanied with decreases in robustness.

In conclusion one can note that the simple SAC behavior introduced here works well, typically achieving 98% *sensory* area coverage, and that it owes its success to its deadlock avoidance procedure, which is frequently used during exploration.

References

1. Acar, E.U., Choset, H.: Sensor-Based Coverage of Unknown Environments. Int. J. Robotics Research 21, 345–366 (2002)
2. Edlinger, T., von Puttkamer, E.: Exploration of an Indoor-Environment by an Autonomous Mobile Robot. In: Proc. of IROS 1994, pp. 1278–1284 (1994)
3. González-Baños, H.H., Latombe, J.-C.: Navigation Strategies for Exploring Indoor Environments. Int. J. Robotics Research 21, 829–848 (2002)
4. Stachniss, C., Burgard, W.: Exploring Unknown Environments with Mobile Robots using Coverage Maps. In: Proc. of IJCAI 2003, pp. 1127–1132 (2003)
5. Zelinsky, A., et al.: Planning Paths of Complete Coverage of Unstructured Environments by a Mobile Robot. In: Proc. of ICAR 1993, pp. 533–538 (1993)
6. Wahde, M.: A General-Purpose Method for Decision-Making in Autonomous Robots. In: Chien, B.-C., Hong, T.-P., Chen, S.-M., Ali, M. (eds.) IEA/AIE 2009. LNCS (LNAI), vol. 5579, pp. 1–10. Springer, Heidelberg (2009)

Topological SLAM Using Fast Vision Techniques

Felix Werner, Frederic Maire, and Joaquin Sitte

School of Information Technology,
Faculty of Science and Information Technology,
Queensland University of Technology,
1 George Street, Brisbane, QLD 4001, Australia
NICTA, Queensland Lab,
Staff House Road, St Lucia, QLD 4072, Australia
{f.werner,f.maire,j.sitte}@qut.edu.au
http://www.scitech.qut.edu.au
http://www.nicta.com.au

Abstract. In this paper we propose a method for vision only topological simultaneous localisation and mapping (SLAM). Our approach does not use motion or odometric information but a sequence of noisy visual measurements observed by traversing an environment. In particular, we address the perceptual aliasing problem which occurs using external observations only in topological navigation.

We propose a Bayesian inference method to incrementally build a topological map by inferring spatial relations from the sequence of observations while simultaneously estimating the robot's location. The algorithm aims to build a small map which is consistent with local adjacency information extracted from the sequence measurements. Local adjacency information is incorporated to disambiguate places which otherwise would appear to be the same.

Experiments in an indoor environment show that the proposed technique is capable of dealing with perceptual aliasing using visual observations only and successfully performs topological SLAM.

Keywords: Autonomous mobile robots, SLAM, correspondence problem, topological navigation, panoramic vision, colour histograms.

1 Introduction

Simultaneous localisation and mapping (SLAM) is one of the most researched areas in robotics. Two different approaches exist to the SLAM problem: *Metric* and *topological* [1]. The fromer aims to model the environment using a metric map so geometrically accurate position estimation is achieved. Topological maps are graphical models of the environment that capture *key places* and their *connectivity* in an abstract and compact manner for localisation and path planning.

In both cases, probabilistic approaches have been successfully applied to deal with the inherent uncertainties associated with robot perception, that would otherwise trouble the map-building process. Beside measurement noise, topological

J.-H. Kim et al. (Eds.): FIRA 2009, LNCS 5744, pp. 187–196, 2009.

mapping is complicated through *perceptual aliasing* which occurs when physically different parts in in the environment appear to be the same to the robot. This phenomenon occurs as sensors may supply insufficient data to identify the current state of the world because of sensory noise, limited field of view (aperture problem) and repeated structures in the environment. Perceptual aliasing makes it difficult for a robot to decide when it is visiting a new place or revisiting a memorised place (*loop closing*) [1,2].

In this work, we address the problem of topological SLAM from a sequence of visual measurements obtained from visited places only. In particular, we are interested in the problem of loop closing in environments which contain physically different places which appear to be the same. We approach this problem using Bayesian inference to estimate the posterior distribution on topological maps while simultaneously determining the place the robot currently occupies. The inference method embeds a strategy to reliably deal with percptual aliasing by distinguishing similar places on the basis of neighbouring information [3]. Using neighbouring information to disambiguate physically different places which appear identical was proposed by Werner et al. [3]. However, it was only examined on artificial topological graphs with deterministic observations.

Our approach is purely vision based so we suppose that actions (e.g. turn left) and odometry cannot be sensed directly. In contrast to other vision based methods [4,5] which aim to represent places highly distinctive using sophisticated visual features such as SIFT [7], or SURF [8,9], we use colour histograms only [10,11,12,13,14]. Colour histograms in conjunction with panoramic images exhibit several attractive properties such as invariance to rotation around the vertical axis. Moreover, colour histograms represent salient colour information of images in a very compact manner and are very fast to extract and process. Clearly, colour histograms are not very distinctive image features so we have the scenario that similar visual appearance is shared by different places.

Research in topological SLAM has mainly been concerned with a particular aspect to avoid the perceptual aliasing problem by improving the distinctiveness of the appearance of places [4,5]. These approaches do not properly address situations where places are indistinguishable even with perfect sensing. Other approaches support the robot's perceptual abilities using metric information gained from odometry measurements or the robot's actions [15,16,17].

2 Neighbourhood Information for Topological Mapping

Our algorithm represents a topological map by a *labelled graph* where *vertices* represent places and *edges* reflect the connectivity between places [1]. The *labels* of vertices refer to *fingerprints* which characterise the place in terms of sensor data. The graph which corresponds to the surrounding is denoted *environment graph* G_{env} and the *map graph*, which we want to infer, G_{map}. The environment graph is unknown and the only available information about it is a finite history $h_{env} = l^0_{env}, l^1_{env}, ... \in L^*_{env}$ of labels of visited vertices obtained from the traversal of the environment graph (Here, * is the Kleene star).

Our method exploits the neighbourhoods of places to disambiguate places with similar appearance. The neighbourhoods of the environment graph are not accessible directly as it is unknown. Local neighbouring information which is contained in the history is accessible through sequences of length n, called n-grams [3]. Consecutively visited vertices are represented by consecutive labels in the history and, in turn, consecutive labels in the history must originate from adjacent vertices in the environment graph. Hence, the set of n-grams $Grams(h, n)$, which can be obtained from a history, corresponds to a *feature space* on the history.

In order to achieve reliable navigation, a robot requires an internal representation that exhibits the properties of the environment with respect to the selected representation (e.g. topological). In our case, a map graph is required to be isomorphic with the environment graph. The map graph is isomorphic with the environment graph if there is a bijective mapping such that each neighbourhood of the map graph corresponds to a neighbourhood of the environment graph and vice versa. However, it is not possible to compare neighbourhoods of the map graph directly with neighbourhoods in environment graph as the latter one is unknown. Consequently, we propose to measure the consistency of graphs in the feature space; that means, the sets of n-grams of the graphs. Hence, two graphs are n-consistent if they share the same set of n-grams [3]. For noisy data the Hausdorff distance is used to measure the n-consistency of two graphs G_0 and G_1 using $\Gamma_0 = Grams(h_0, n)$ and $\Gamma_1 = Grams(h_1, n)$ generated from $h_0 \in L_0^*$ and $h_1 \in L_1^*$, so

$$d_H(\Gamma_0, \Gamma_1) = \max(\max_{\gamma_i \in \Gamma_0} \min_{\gamma_j \in \Gamma_1} d(\gamma_i, \gamma_j), \max_{\gamma_j \in \Gamma_1} \min_{\gamma_i \in \Gamma_0} d(\gamma_j, \gamma_i)). \tag{1}$$

The smaller the Hausdorff distance the more n-consistent are graphs G_0 and G_1. The distance of two n-grams γ_i and γ_j is computed using the maximum norm

$$d(\gamma_i, \gamma_j) = ||\gamma_i - \gamma_j||_\infty = \max_{k=0\ldots n-1}(|\gamma_{i,k} - \gamma_{j,k}|) \tag{2}$$

so the distance between two sets of n-grams is determined by the most significant distance of two labels which are mapped to the same vertex.

3 Topological SLAM

In this section we describe our method for topological SLAM from a sequence of visited places. The history of observations of visited places is the only information about the environment. In particular, the robot has no access to metric information such as odometry and no information about its actions but is aware of performed U-turns. We suppose that the robot has explored the environment and has recorded a history $h_{env} = l_{env}^0, ..., l_{env}^{M-1}$ of M fingerprints of visited places. After the exploration run the set $\Gamma_{env} = Grams(h_{env}, n)$ is derived from the history.

3.1 Bayesian Map Inference

The space of topologies grows hyper-exponentially with the number of measurements [15]. Thus, we use a Bayesian inference method to only infer topological maps which are consistent with the observations. In Bayesian map inference observations are used to update or to newly infer the probability that a hypothesis may be true using Bayes' theorem. For the purpose of inferring a topological map graph G_{map} from a history h_{env} we have

$$P(G_{map}|h_{env}) \propto P(h_{env}|G_{map})P(G_{map}). \tag{3}$$

We can assume the process of incrementally building a topological map is Markovian – that is the current topological map contains all relevant information and is conditionally independent of all earlier states. Consequently, we write Equation 3 in an incremental way

$$P(G_{map}^{t+1}|l_{env}^{0:t+1}) \propto P(l_{env}^{t+1}|G_{map}^{t})P(G_{map}^{t}) \tag{4}$$

to estimate the posterior distribution on topological maps $P(G_{map}^{t+1}|l_{env}^{0:t+1})$ from the prior distribution $P(G_{map}^{t})$ using the measurement likelihood $P(l_{env}^{t+1}|G_{map}^{t})$. It is difficult to represent uncertainty directly in a topological map so we use a sequential Monte-Carlo technique to represent uncertainty by maintaining a collection of N map candidate samples which are randomly drawn from the probability density function in the space of topological environment maps. A collection of map samples $\left\{G_{map,i}^{t+1}, w_i^{t+1}\right\}_{i=0}^{N-1}$ is used to model the posterior distribution $P(G_{map}^{t+1}|h_{env}^{0:t+1})$ on topological maps. The weights w_i^{t+1} are importance factors which are normalised such that

$$\sum_{i=0}^{N-1} w_i^{t+1} = 1. \tag{5}$$

Bayesian filters recursively estimate the posterior distribution $P(G_{map}^{t+1}|l_{env}^{0:t+1})$ from the proposal distribution $P(G_{map}^{t})$ and the perception l_{env}^{t+1} using two distinct phases: Prediction and Update.

The prediction phase uses the map estimate from the previous time step to estimate the map at the next time step. In our case, we can not predict the next observation as the map is not known in advance so the probability which vertex corresponds to the place the robot visits next is uniformly distributed over the vertices contained in the map and an additional vertex which refers to a new place. Therefore, for each sample $G_{map,i}^{t}$ of the collection of samples that models the posterior distribution $P(G_{map}^{t}|h_{env}^{0:t})$ at t, map graph candidates $\left\{\widetilde{G}_{map,i,k}^{t+1}\right\}_{k=0}^{|V_{map,i}|}$ are generated where k refers to the vertex which the new observation is predicted to correspond to. Gaussian white noise $\mathcal{N}(0, \sigma_g)$ is added artificially to the labels of the map graph candidates to model the inherent uncertainties associated with robot perception. If an additional vertex is

introduced it is labelled with the new observation. If the current location and the place which is predicted to be occupied next by the robot are not connected in the topological map yet, an edge is introduced to connect the corresponding vertices.

The proposal distribution on topological maps is updated using the new observation. In Monte-Carlo approaches this is done by weighting the samples using the data. We compute the weight of a topological map graph candidate $\widetilde{G}^{t+1}_{map,i,k}$ using the current observation l^{t+1}_{env} and the set of n-grams Γ_{env}, so

$$w_i = P(l^{t+1}_{env}, \Gamma_{env} | \widetilde{G}^{t+1}_{map,i,k}) = P(l^{t+1}_{env} | \widetilde{G}^{t+1}_{map,i,k}) P(\Gamma_{env} | \widetilde{G}^{t+1}_{map,i,k}) P(\widetilde{G}^{t+1}_{map,i,k}). \quad (6)$$

The term $P(l^{t+1}_{env} | \widetilde{G}^{t+1}_{map,i,k})$ computes the probability of the measured label and the label of the vertex k to be identical, so

$$P(l^{t+1}_{env} | \widetilde{G}^{t+1}_{map,i,k}) = \exp\left(- \left(\frac{l^{t+1}_{env} - \tilde{l}_k}{\sigma_l} \right)^2 \right) \quad (7)$$

where $\tilde{l}_k \in \widetilde{L}^{t+1}_{map,i,k}$ and σ_l denotes a weighting factor.

The second term in Equation 6 considers the probability that map graph $\widetilde{G}^{t+1}_{map,i,k}$ is n-consistent with the information given in the history. Using Equation 1 the consistency probability is computed with

$$P(\Gamma_{env} | \widetilde{G}^{t+1}_{map,i,k}) = \exp\left(- \left(\frac{d_H(\Gamma_{env}, \widetilde{\Gamma}^{t+1}_{map,i,k})}{\sigma_c} \right)^2 \right) \quad (8)$$

where $\widetilde{\Gamma}_{map,i,k} = Grams(\tilde{h}^{t+1}_{map,i,k}, n)$ is generated from $\tilde{h}^{t+1}_{map,i,k} \in \widetilde{L}^{*,t+1}_{map,i,k}$.

In parametric methods the probability of the model to represent the data is increased when the number of parameters in the model is increased. In our case, a map graph which consists of one component for each n-gram in $Grams(h_{env}, n)$ would be consistent with the information from the history but is inappropriate for navigation, containing too many vertices. Hence, the prior should favour small topological maps. Consequently, the last term in Equation 6 penalises map graphs which contain vertices with similar labels

$$P(\widetilde{G}^{t+1}_{map,i,k}) = \prod_{a=0}^{|\widetilde{V}^{t+1}_{map,i,k}|} \prod_{b=0}^{|\widetilde{V}^{t+1}_{map,i,k}|} \left(1 - \phi \left(\exp - \left(\frac{\tilde{l}_a - \tilde{l}_b}{\sigma_l} \right)^2 \right) \right) \quad (9)$$

where $\tilde{l}_a, \tilde{l}_b \in \widetilde{L}^{t+1}_{map,i,k}$ and ϕ weights the influence of the penalty.

The posterior distribution on topological maps is computed by drawing N samples from the proposal distribution.

3.2 Localisation

While building the map, the place the robot occupies is implicitly estimated whenever the map graph is updated with a new observation. The vertex whose

label is updated or additionally introduced using the observation indicates the estimated location of the robot. If a new vertex is introduced the robot is hence located at the place which corresponds to that vertex. The location, in turn, is used to guide the mapping process by introducing adjacencies between the current and the previous place occupied.

4 Experiments

Our experimental set up environment covers an indoor office environment area of about 20,000 square meters (Wean Hall at Carnegie Mellon University), see Figure 1. The robot uses a panoramic camera to acquire information about the environment.

The experimental platform traverses the environment using the generalised Voronoi graph (GVG) strategy developed by Choset and Nagatani [16]. It is based on the Voronoi diagram which is a special kind of decomposition of a metric space into segments and nodes determined by distances to a specified discrete set of objects in the space. Our robot measures distances using sonar readings.

Segments capture the points in the plane that are equidistant to two sites. Travelling along the Voronoi segments, the robot can keep in the middle of corridors while exploring the environment. The Voronoi nodes are the points equidistant to three (or more) obstacles. In indoor environments this corresponds naturally to T-junctions or intersections of corridors as shown in Figure 1. Once such a locus point is identified, a panoramic image is taken.

Note, we use the GVG strategy only for exploring the environment and the identification of places. Mapping and localisation is performed using purely visual information. The GVG of our experimental environment is displayed in Figure 1.

In some areas of our experimental environment bright ceiling lights are installed whereas some other areas have wall lights. As a result, the images the robot takes at each place suffer from loss of clarity of visual information within shadows or near strong lights (over and under exposed regions). We use a method

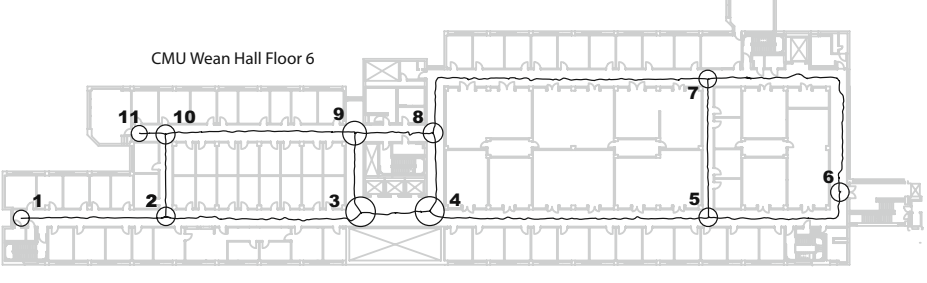


Fig. 1. The floor plan of Wean Hall Floor 6 at Carnegie Mellon University (CMU). Embedded is the topological graph (vertices and their connectivity) that reflects the ground truth of the topological map that we wish the SLAM algorithm to infer.

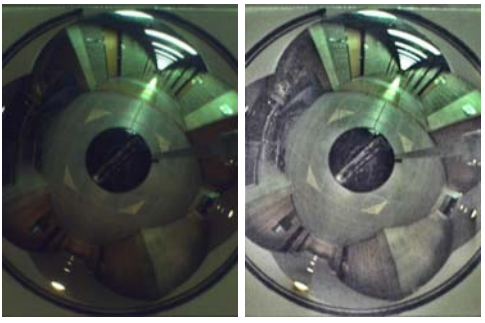


Fig. 2. An under exposed (left) image measured at place 10 in Wean Hall (see Figure 1) and the enhancement (right) using Vonikakis and Andreadis method [18].

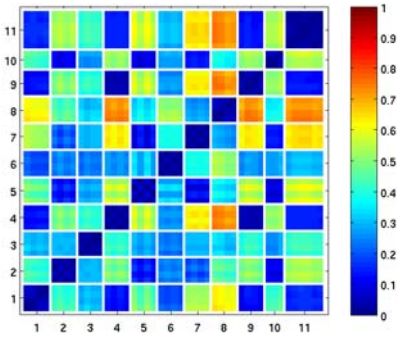


Fig. 3. Distance matrix of the fingerprints of places (see Figure 1). Blue small distance, red big distance.

proposed by Vonikakis and Andreadis to enhance the acquired images by lightening under exposed regions and darkening over exposed regions without affecting the correctly exposed ones [18]. Figure 2 shows the application of the enhancement method for an under exposed image.

In our system, the visual appearance of a place is measured using colour histograms. Usually a colour histogram is created by calculating an N-bin histogram for each of the R, G and B colour bands and so looses the 3D spatial information of the RGB tuples in colour space. To retain the 3D spatial information of the RGB tuples in colour space, we use 3D histograms in RGB space where the histogram consists of N^3 equally sized bins.

The number of n-grams we can extract at most from a history h of lenth m is at most $m - n + 1$ whereby the maximum number of unique n-grams which are derivable from a strongly connected graph is $O(|L|^n)$ [3]. However, using fast vision techniques such as 3D colour histograms with 5^3 bins keeps the system fast despite potentially big sets of n-grams. In comparison, a single standard SIFT feature as is represented through a 128 dimensional vector, whereby thousands of such features may be identified in a single image [7].

We have conducted several exploration runs and recorded a total of 50 images of places and stored the fingerprints of the enhanced images in a data base. Figure 3 displays the similarity matrix of the fingerprints from the enhanced images. It is apparent that the environment contains numerous topological ambiguities when using colour histograms as fingerprints of the places.

Given the data base and the ground truth environment graph, we can simulate arbitrary traversals of the environment. The robot starts at an initial vertex and selects an arbitrary adjacent vertex as next place. According to the vertex the robot occupies, a random observation from the data base is sampled. For the following evaluations, 2×200 paths of length 70 were generated. One set of paths uses fingerprints from the enhanced images and the other set uses fingerprints from the original images. Each path represents an exhaustive exploration of the

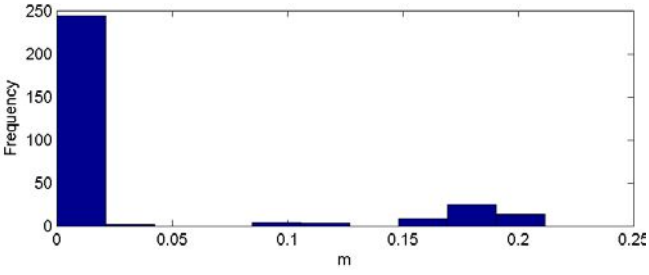


Fig. 4. The consistency measured using the Hausdorff distance of the inferred topological maps and the corresponding simulated histories is displayed. Most of the inferred maps using the enhanced visual perceptions are highly consistency respective the consistency measure.

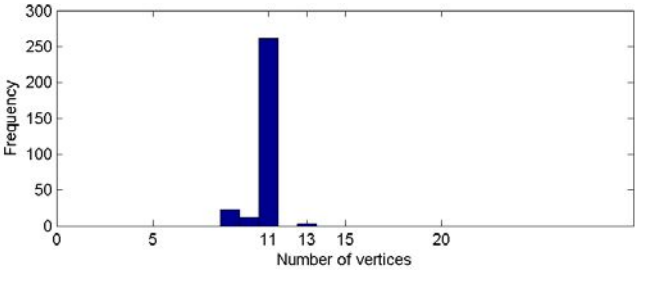


Fig. 5. The number of vertices of the inferred topological maps are histogrammed. Most of the inferred maps have only small consistency errors which occur due to measurement noise. A clear peak at 11 vertices is to see what corresponds to the environment graph (see Figure 1).

environment. For each path the set Γ_{env} of n-grams is derived before starting the algorithm. Note, it is actually not necessary to assume an exhaustive exploration of the environment as the inferred map is a representation of the environment which is consistent to the measurements at a certain time.

We have applied the proposed SLAM method using 30 map candidates to model the posterior distribution on topological maps and 3-consistency mapping. The fingerprints of places are represented through 3D colour histograms with 5^3 bins from the enhanced images.

Figure 4 shows a histogram of the 3-consistency of the inferred maps of the simulated random traversals. It can be seen that most of the inferred maps using the enhanced images are very consistent with the information from the history, whereby little divergence occurs as a result of measurement noise inherent to sensor perception. Rare outliers may occur when the mapping process is misled due its probabilistic nature so the inferred map is inconsistent with the observations.

The overall goal in topological mapping is to build an internal representation which is isomorphic to the environment. Here, we investigate whether the

inferred map graphs are isomorphic to the environment graph in order to measure the quality of the proposed approach for topological mapping. In the case of applying the image enhancement method, we found all map graphs (193, or 97%) with the same number of vertices as the environment graph to be isomorphic to the environment graph (see Figure 5). The results support the strategy of the proposed algorithm to use the current position estimation with the new observation to map the connectivity of the environment.

5 Discussion

In this paper we proposed a Bayesian approach for topological SLAM that does not rely on any motion model or metric information, but uses a history of noisy visual measurements from visited places only. Using colour histograms as fingerprints of places makes our system fast but entails physically different places to appear similar to the robot's senses. In order to deal with this problem, the sequential Monte-Carlo SLAM technique embeds a method to reliably disambiguate places which appear to be the same but in fact are different. The method aims to maintain consistency with the observed data while minimising the number of vertices contained in the map. The consistency between a topological map and the observations is measured using the Hausdorff distance.

Experiments in an indoor environment which is subject to severe ambiguities due to repeated structures demonstrate the capability of the idea to use neighbourhood clues in order to disambiguate otherwise identical vertices. Our approach mostly infers topological maps with only small inconsistencies with respect to the data. Moreover, most of the resulting maps are isomorphic to the environment graph what supports reliable topologial navigation despite severe perceptual aliasing.

Acknowledgements

NICTA is funded by the Australian Government as represented by the Department of Broadband, Communications and the Digital Economy and the Australian Research Council through the ICT Centre of Excellence program.

Computational resources and services used in this work were provided by the HPC and Research Support Group, Queensland University of Technology, Brisbane, Australia.

The authors would like to thank Stephen Tully for the help with the experiments and Howie Choset for valuable discussions.

References

1. Filliat, D., Meyer, J.A.: Map-based navigation in mobile robots: I. a review of localization strategies. Cog. Sys. Res. 4(4), 243–282 (2003)
2. Thrun, S., Burgard, W., Fox, D.: Probabilistic Robotics. MIT Press, Cambridge (2005)

3. Werner, F., Gretton, C., Maire, F., Sitte, J.: Induction of topological environment maps from sequences of visited places. In: IROS, Nice, France, pp. 2890–2895. IEEE, Los Alamitos (2008)
4. Angeli, A., Doncieux, S., Meyer, J.A., Filliat, D.: Incremental vision-based topological slam. In: IROS, September 2008, pp. 1031–1036 (2008)
5. Valgren, C., Duckett, T., Lilienthal, A.: Incremental spectral clustering and its application to topological mapping. In: ICRA, Roma, Italy, pp. 4283–4288. IEEE, Los Alamitos (2007)
6. Cummins, M., Newman, P.: Fab-map: Probabilistic localization and mapping in the space of appearance. Int. J. Rob. Res. 27(6), 647–665 (2008)
7. Lowe, D.: Distinctive image features from scale-invariant keypoints. In: IJCV, vol. 60, pp. 91–110. Kluwer Academic Publishers, Dordrecht (2003)
8. Bay, H., Tuytelaars, T., Van Gool, L.: SURF: Speeded up robust features. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006. LNCS, vol. 3951, pp. 404–417. Springer, Heidelberg (2006)
9. Tuytelaars, T., Mikolajczyk, K.: Local invariant feature detectors: A survey. FnT Comp. Graphics and Vision 3(3), 177–280 (2008)
10. Werner, F., Sitte, J., Maire, F.: Visual topological localisation and mapping using colour histograms. In: ICARCV, Hanoi, Vietnam. IEEE, Los Alamitos (2008)
11. Werner, F., Sitte, J., Maire, F.: Automatic place determination using colour histograms and self-organising maps. In: ICAR, Jeju, Korea Republic, August 2007, pp. 111–115 (2007)
12. Werner, F., Sitte, J., Maire, F.: On the induction of topological maps from sequences of colour histograms. In: DICTA, Adelaide, Australia, pp. 167–174. IEEE, Los Alamitos (2007)
13. Ulrich, I., Nourbakhsh, I.R.: Appearance-based place recognition for topological localization. In: ICRA, pp. 1023–1029 (2000)
14. Rañó, I., Lazkano, E., Sierra, B.: On the application of colour histograms for mobile robot localisation. In: EMCR, vol. 1, pp. 189–193 (2005)
15. Ranganathan, A., Dellaert, F.: Inference in the space of topological maps: An mcmc-based approach. In: IROS, Sendai, Japan, vol. 2, pp. 1518–1523. IEEE, Los Alamitos (2004)
16. Choset, H., Nagatani, K.: Topological simultaneous localization and mapping (slam): Toward exact localization without explicit localization. IEEE Trans. Robot. Automat. 17(2), 125–137 (2001)
17. De, A., Lee, J., Cowan, N.J.: Toward slam on graphs. In: WAFR, Guanajuato, México (December 2008)
18. Vonikakis, V., Andreadis, I.T.: Fast automatic compensation of under/Over- exposured image regions. In: Mery, D., Rueda, L. (eds.) PSIVT 2007. LNCS, vol. 4872, pp. 510–521. Springer, Heidelberg (2007)

Virtual Door-Based Coverage Path Planning for Mobile Robot

Hyun Myung[1,⋆], Hae-min Jeon[1], Woo-Yeon Jeong[2], and Seok-Won Bang[3]

[1] Department of Civil and Environmental and Engineering, KAIST Yuseong-gu, Daejeon, 305-701, Republic of Korea
hmyung@kaist.ac.kr, inhishand@kaist.ac.kr

[2] Samsung Advanced Institute of Technology, Samsung Electronics, Co. Ltd., Giheung,Yongin, 449-712, Republic of Korea
wooyeon.jeong@samsung.com

[3] Robotics Institute, Carnegie Mellon Univ. 5000 Forbes Avenue, Pittsburgh, PA 15213, USA

Abstract. This paper presents a novel coverage path planning method in indoor environment for a mobile robot such as cleaning robot. Overall region is divided into several sub-regions based on the virtually extracted doors. The algorithm is inspired from the usual way of dividing an indoor environment into sub-regions, i.e., rooms based on the identification of doors. The virtual door algorithm extracts the virtual doors by combining a Generalized Voronoi Diagram (GVD) and a configuration space eroded by the half of the door size. The region to region cleaning algorithm is also proposed based on the closing and opening operations of virtual doors. The performance of the proposed algorithm has been tested on various real indoor environments using a commercially available cleaning robot.

Keywords: Virtual Door; Coverage Path Planning; Mobile Robot.

1 Introduction

The purpose of coverage path planning is to make the robot visit all reachable area in a given environment efficiently. Various approaches have been proposed to deal with coverage path planning problems. One of the most well known approaches is the utilization of a divide and conquer strategy [1]. In this approach, a given environment is divided into several regions called cells, and each cell is covered by simple robot motion. Choset [2] introduced a coverage method based on an exact cellular decomposition called boustrophedon decomposition. In this method, a line segment called a slice is swept through a given environment. When there is a change in the connectivity of the slice, a new cell is formed, and the cell is then described by points termed critical points. After cell decomposition, the robot covers each cell using a simple zigzag motion called boustrophedon

⋆ Assistant Professor, Ph.D., Adjunct Prof. of KIUSS(KAIST Institute for Urban Space & Systems) and Robotics Program.

J.-H. Kim et al. (Eds.): FIRA 2009, LNCS 5744, pp. 197–207, 2009.

pattern. The disadvantage of this approach is that it divides the area in only one direction.

Acar et al. [3], [4] presented a framework for coverage task using Morse decomposition. In this method, the critical points of Morse function are used to form the cells. Using the property that the topology of the slice remains constant between the critical points, this method easily guarantees complete coverage of a cell. Wong and MacDonald [5], [6] proposed a cell decomposition algorithm based on topological structures of the robot's environment. This form of decomposition, termed slice decomposition, uses changes in topology to decompose and environment, where each cell intersects with the sweep line twice as it passes over. In this method, the cell decomposition results are represented as a topological map, and each cell is covered by the robot via simple zigzag motion. These divide and conquer approaches are very powerful in that completeness of coverage can be easily achieved by using simple algorithms to cover each region.

When task efficiency is considered, time efficiency as well as completeness of coverage is important factors. A robot should cover a given area as fast as possible. Some previous researches have dealt with the efficiency of the coverage. Huang [7] focused on cell decomposition itself to achieve optimal coverage. In this method, the criterion for optimization is the total number of turns required to cover all sub-regions. For reducing the number of turns, a slice direction that minimizes the sum of the altitudes of all cells is chosen. Huang suggested an algorithm that performs multiple line sweeps to decompose the given environment into cells and applied dynamic programming to combine cells into larger sub regions and to assign a sweep direction for each sub-region. Although this approach is limited to polygonal environments and requires a great deal of time for cell decomposition, it considers efficiency of coverage within the divide and conquer approach. Kang et al. [8] attempted to achieve optimal coverage by minimizing extra relocation moves between cells. The main concept of this method is ensuring that entry and exit points for various sub-regions of a given environment are in close proximity. However, this approach do not consider the border of rooms such as doors, therefore the coverage of the robot is not limited in a certain area and can go back and forth between rooms.

In this paper, a novel coverage path planning algorithm is proposed utilizing virtual door concept. The robot is assumed to be operated in an enclosed indoor environment and knows its map such as an occupancy grid or coarse map. Our method basically achieves complete coverage by adopting divide and conquer strategy. The algorithm decomposes a given area into sub-regions based on the location of the virtual doors. During the cell decomposition process, small and narrow cells are merged into larger neighbor cells. After the cell decomposition, the optimal coverage direction is chosen between vertical and horizontal ways for each sub-region. This optimal direction reduces the time for coverage as the number of rotation movement is decreased.

A virtual wall device, which is similar to our path planning algorithm using virtual doors, has been applied to a cleaning robot named Roomba produced by iRobot Company. However, it uses artificially installed infrared sensors to form

a virtual wall by emitting beams. On the other hand our algorithm doesn't need any sensors or any extra equipment for generating robot's path.

The remainder of this paper is organized as follows. Section 2 describes the cell decomposition algorithm using virtual doors for a given environment. Some experimental results are presented in Section 3. Conclusions will be discussed in Section 4.

2 Virtual Door Algorithm

A flow chart of virtual door algorithm is shown in Fig 1. As shown in Fig. 1, the main idea of this algorithm is dividing whole area into sub-regions based on virtual doors. This concept is from the way of how people conceptually divide the whole area in enclosed indoor environments. Many people usually divide the area into several rooms for example, living room, bed room and kitchen and this separation is normally based on walls or doors. According to this concept, the virtual door algorithm is proposed to calculate the possible locations where the doors might exist. The doors can be extracted by the following process shown in Fig 1.

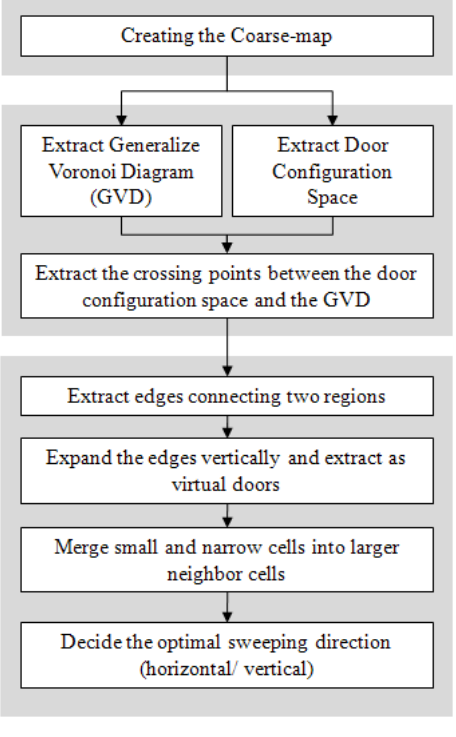


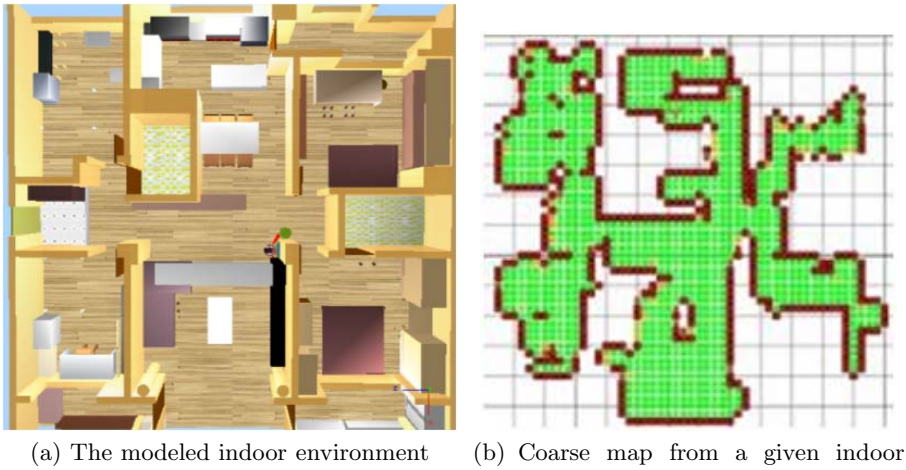
Fig. 1. Virtual door extraction algorithm

2.1 Preprocessing

A robot is assumed to be operated in an enclosed indoor environment such as shown in Fig. 2(a) and knows its map before coverage. Especially in this paper, a coarse map is obtained using SLAM(Simultaneous Localization And Mapping) technique while the robot moves at the first time as shown in Fig. 2(b). The coarse map which is used in this experiment is consisted of cells of the same size of a mobile robot's inlet. This coarse map reduces the computational burden of path generation as the size of each cell is increased and it cause the total number of cells to calculate in given area is decreased. After mapping the environment, the door configuration space is constructed and the Generalized Voronoi Diagram based on the coarse map. The configuration space is originally the map that is obtained by eroding as much as half of a robot size from a given grid map. However, in this paper the door configuration space is generated by eroding as much as half of the size of the door. The door configuration space is shown in Fig. 3(a). The Generalized Voronoi Diagram is a road map method that tends to maximize the distance between the robot and obstacles in the map. This method can be used to deal with collision avoidance problem which is one of the important factors when the path is generated. The Generalized Voronoi Diagram based on the simulated environment is shown in Fig. 3(b). Both door configuration space and the Generalized Voronoi Diagram can be easily obtained using morphological operation in the image processing.

2.2 Virtual Door Extraction

From the obtained door configuration space and Generalized Voronoi Diagram, the crossing points are extracted by overlapping two images. Among the crossing



(a) The modeled indoor environment (b) Coarse map from a given indoor environment

Fig. 2. Simulated indoor environment and its coarse map

(a) Door configuration Space

(b) Generalized Voronoi Diagram obtained from the coarse map.

(c) Overlapping GVD and door configuration space

(d) Space connectivity diagram

(e) Extracted edges after elongation in vertical and horizontal direction

(f) Virtual doors after merging small regions into larger neighboring subregions

Fig. 3. Application of door extraction algorithm to simulated indoor environment

points, only the points are selected that are connects two sub-regions. To know whether the crossing edges exist between two sub-regions, we check the condition of the space connectivity as shown in Fig. 3(d). Based on both images in Fig. 3(c) and Fig. 3(d), the virtual doors can be extracted after extending the edges vertically or horizontally until it reaches the obstacles in both sides to separate the regions. The extended edges are shown in Fig. 3(e). At the final step, the small or narrow spaces are merged into a larger neighbor cells. This merging process reduces the computational burden of path generation as the number of sub regions to cover is decreased. The virtual doors after merging minor regions are shown in Fig. 3(f).

The finally obtained virtual doors are shown in the original coarse map in Fig. 4. As can be seen in Fig. 4, virtual doors include real doors. The unmatched virtual doors indicate the size of the region's entrance was much like that of the real doors.

After decomposing cells by virtual door algorithm, the optimal path direction either horizontal or vertical is calculated for each sub-region. The normal boustrophedon decomposition algorithm can divide and generate paths in one direction in accordance with a sweep line direction but the virtual door algorithm can determine an efficient path direction like that of Huang's algorithm [7]. The number of crossing points is calculated between edges using equally divided sweep lines. Because a direction which has smaller number of crossing points is the direction that needs less number of rotations, the direction which has less number of crossing points is selected as the optimal path direction.

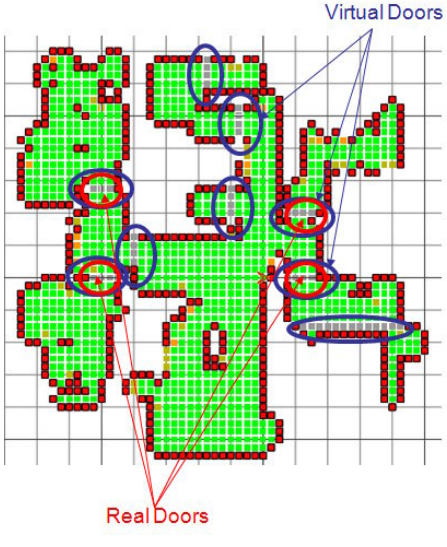


Fig. 4. Extracted virtual doors using the virtual door algorithm

2.3 Region to Region Cleaning Algorithm

After calculating the optimal cleaning direction for each sub-region, the robot can perform the region-based cleaning. When the robot starts cleaning, all the virtual doors are closed to restrict the cleaning in the starting region. And when the robot determines the sub-region has been completely covered, the robot moves to another sub-region by opening all virtual doors. When all virtual doors are open, the robot calculates the optimal path to the nearest uncovered area. If the path exists, the robot moves to that region by following the calculated path and see whether the robot has reached to another region. When the robot decides it has reached to another region, then it closes all virtual doors. Then the same routine can be followed as can be seen in Fig. 5.

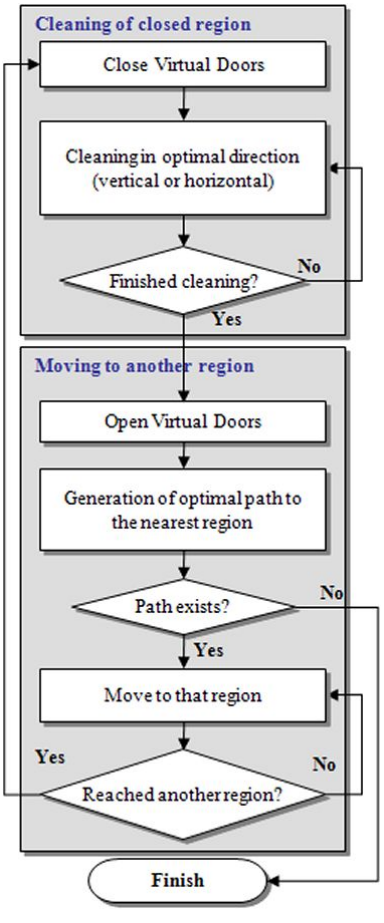


Fig. 5. Region by region cleaning algorithm by applying closing and opening operation of virtual doors

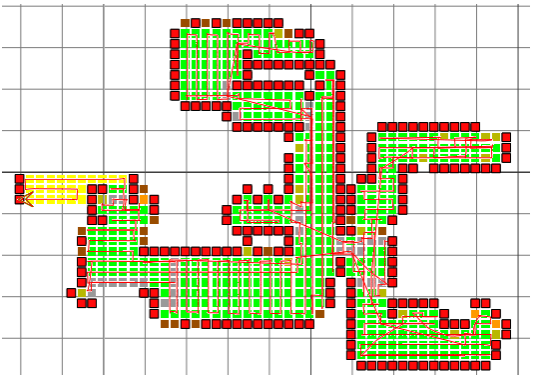


Fig. 6. The robot follows the planned path after virtual door extraction from the simulated environment

Fig. 6 shows the path generated by the region by region cleaning algorithm for the simulated environment shown in Fig. 2(a) and Fig. 4. The robot follows the optimal cleaning direction of each sub-region by closing and opening virtual doors.

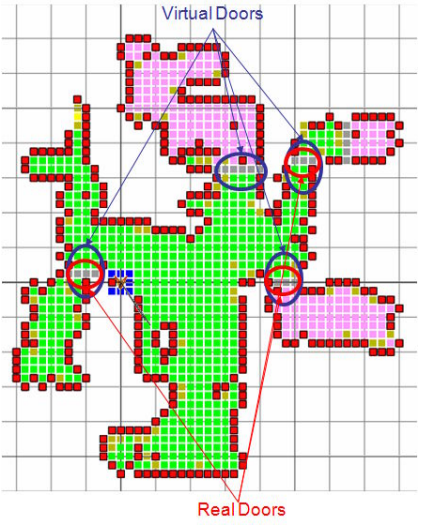
3 Experimental Results

In order to verify the proposed algorithm for coverage path planning real experiments have been carried out in various real environments. Two specific results will be discussed in this section: one for an apartment ($158m^2$) in Seoul, Korea and another for an apartment ($107m^2$) in Yongin, Korea.

In our scenario, a mobile robot is used for mapping the given environment and after the mapping process our algorithm is used for calculating the possible locations of virtual doors. Within this scenario, a vacuum cleaning robot VC-RE70V developed by Samsung Electronics Co. Ltd. (Fig. 7) is used and our algorithm works both in two different spaces. VC-RE70V has a camera on top

Fig. 7. Samsung cleaning robot VC-RE70V used in the experiment

(a) Floor plan of the apartment



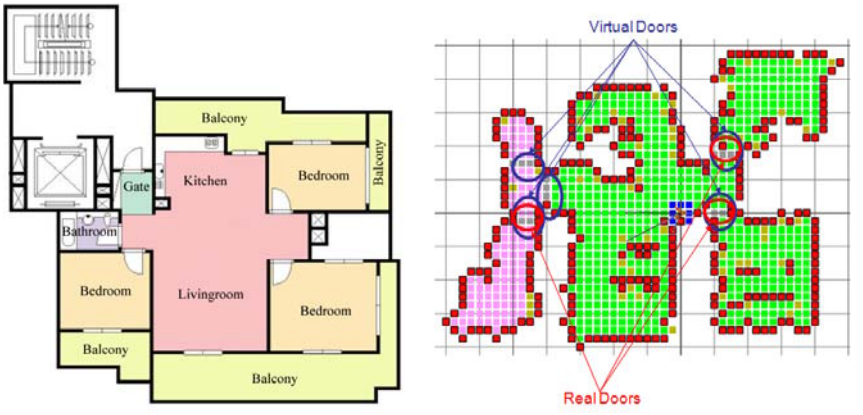
(b) Comparison of virtual doors (gray-colored and blue-circled) and real doors (red-circled). Pink cells indicate the horizontal cleaning direction and the green cells indicate the vertical cleaning direction

Fig. 8. The result of coverage path planning using virtual door algorithm in an apartment in Seoul, Korea

of the robot's body and can localize itself using the vision-based SLAM (Simultaneous Localization And Mapping) algorithm and vision information obtained from the camera. The map-building of the environment is done by short range (less than 1 meter) IR (Infra-Red) distance sensors installed in front of the robot.

From the real world applications, our algorithm was seen to work well even in dynamic environments where unknown and moving obstacles exist. The results are shown in Figs. 8 and 9.

In Fig. 8, the number of virtual doors extracted is four, while the number of real doors is three. Most of the virtual doors coincide with the real doors but one. At the center of the map, the virtual door that is not the real door was extracted because the size of the region's entrance was much like that of the door. According to the number of virtual doors, the sub-regions are determined and the optimal cleaning direction can be set different for each sub-region. Thus the exceeded number of virtual doors does not have bad effect on the overall cleaning time. The regions with pink rectangles indicate the horizontal cleaning direction and the green regions indicate the vertical cleaning direction. In Fig. 9, the number of extracted virtual doors is five while the number of real doors is three. The figure also indicates the optimal cleaning direction for each sub-region.



(a) Floor plan of the apartment

(b) Comparison of virtual doors (gray-colored and blue-circled) and real doors (red-circled). Pink cells indicate the horizontal cleaning direction and the green cells indicate the vertical cleaning direction

Fig. 9. The result of coverage path planning using virtual door algorithm in an apartment in Yongin, Korea

4 Conclusion

In this paper, a novel coverage path planning method is proposed using the concept of virtual doors for complete coverage operation of a mobile robot. To cover all the area efficiently and quickly, the virtual door algorithm is proposed. The virtual door algorithm divides the whole given area into sub-regions by virtual doors which are obtained from combining the Generalized Voronoi Diagram and the configuration space images eroded by half of the door size. By closing and opening virtual doors, it is possible to cover a sub-region after a sub-region. Since the robot calculates the optimal direction either horizontal or vertical in each sub-region, the robot can move more efficiently in shorter time. The experimental results using a cleaning robot showed the efficiency of the algorithm compared to the conventional methods.

Acknowledgment

The experiments have been supported by the Robot Navigation Team in Samsung Advanced Institute of Technology, Samsung Electronics Co. Ltd.

References

1. Choset, H.: Coverage for robotics - A survey of recent results. Annals of Mathematics and Artificial Intelligence 31(1-4), 113–126 (2001)
2. Choset, H.: Coverage of known spaces: the Boustrophedon cellular decomposition. Autonomous Robots 9(3), 247–253 (2000)
3. Acar, E.U., Choset, H., Rizzi, A.A., Atkar, P.N., Hull, D.: Morse Decompositions for Coverage Tasks. The International Journal of Robotics Research 21(4), 331–344 (2002)
4. Choset, H., Acar, E., Rizzi, A.A., Luntz, J.: Exact cellular decomposition in terms of critical points of Morse functions. In: Proc. of the IEEE International Conference on Robotics and Automation, San Francisco, USA, pp. 2270–2277 (2000)
5. Wong, S.C., MacDonald, B.A.: A topological coverage algorithm for mobile robots. In: Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems, Lag Vegas, USA, pp. 1685–1690 (2003)
6. Wong, S.C., MacDonald, B.A.: Complete coverage by mobile robots using slice decomposition based on natural landmarks. In: Zhang, C., Guesgen, H.W., Yeap, W.-K. (eds.) PRICAI 2004. LNCS, vol. 3157, pp. 683–692. Springer, Heidelberg (2004)
7. Huang, W.H.: Optimal line-sweep-based decompositions for coverage algorithms. In: Proc. of the IEEE International Conference on Robotics and Automation, Seoul, Korea, pp. 27–32 (2001)
8. Kang, J.W., Kim, S.J., Chung, M.J., Myung, H., Park, J.H., Bang, S.W.: Path Planning for Complete and Efficient Coverage Operation of Mobile Robots. In: Proc. of the 2007 IEEE International Conference on Mechatronics and Automation, Harbin, China, pp. 2126–2131 (2007)

Vision Module for Mini-robots Providing Optical Flow Processing for Obstacle Avoidance

Teerapat Chinapirom[1], Ulf Witkowski[2], and Ulrich Rueckert[1]

[1] System and Circuit Technology, Heinz Nixdorf Institute, University of Paderborn
Fuerstenallee 11, 33102 Paderborn, Germany
[2] Electronics and Circuit Technology, South Westphalia University of Applied Sciences
Lübecker Ring 2, 59494 Soest, Germany
chinapir@hni.upb.de, witkowski@fh-swf.de

Abstract. This paper presents a flexible prototyping platform that can be efficiently used for vision systems of small mobile robots. The vision module has been integrated into the mini-robot "Khepera". The module is utilized to realize optical flow algorithm for obstacle avoidance. The obstacles are detected from abrupt change of the normal flow vectors during operation. This technique is also inspired by visual perception of insects, which alert when an object suddenly appears nearby them. The optical flow algorithm implemented for this approach is Sum of Absolute Differences (SAD) algorithm. The SAD is programmed using the hardware description language (VHDL) efficiently utilizing the FPGA device that is the central processing device of the module. The 30x16 pixels used in SAD for block matching are computed in parallel by 16 pairs of pixels in each operation, which allows in real-time operation. Therefore, the mini-robot being equipped with our 2D vision module for the real-time image processing is able to drive autonomously without collision with obstacles, called ego-motion. The result also shows that the implementation can reduce the execution time compared to serial implementation and helps to reduce energy consumption.

1 Introduction

Modern mobile robots are equipped with various different sensors and many complex actuators more than before. The increasing components cause more and more tasks for a CPU to process the enormous number of sensor's data and handle the actuators, for example, a recent image sensor provides many mega-pixels of data. If a robot possesses a main CPU to process all tasks, the CPU may be overloaded and doesn't achieve some real-time tasks. To prevent the overloaded problem we propose to split the entire processing tasks and then process them by using small and efficient processing units in/nearby the sensors and the actuators. On the one hand, the main CPU is relieved from enormous complex tasks. On the other hand the performance of entire computing system is increasing and effective, for example an embedded FPGA module beside an image sensor [1]. The accelerative (pre)processing part, which is implemented into the FPGA module, filters unnecessary image data and processes the

J.-H. Kim et al. (Eds.): FIRA 2009, LNCS 5744, pp. 208–219, 2009.

relevant data. After all processes are done, the module provides only small symbolic data to the main CPU. Therefore the main CPU is able to run others tasks.

Several robotic vision algorithms are inspired and imitated by biological animals, for example, the visual perception of a fly. The fly has the simple perception, and uncomplicated nerve system, which need very low-power; but effective and sufficient for control flying [2] and for detection any fast-moving objects at near distance before being hit. Some high-class animals are using stereo vision by combining two scenes of left and right eye. This enables e.g. calculation of depth information in the scenes, but it usually requires huge processing resources including the complex neural networks.

An image processing algorithm for detection of moving objects on scenes is the optical flow [3]. This algorithm can be efficiently implemented on mobile systems equipped with a monocular camera. The sequence of grabbed images by the camera will be evaluated for detecting an obstacle and its distance in the processing module. Thus, the robot will be able to move autonomously and to avoid hitting obstacles, so called "ego-movement". The same approach was found in the natural navigation maneuver of bees [4]. Our proposed technique by filtering the grabbed images and extracting regions of interest in the image plane can be used without analysis of all image details. Thus, the necessary algorithms can be applied close to the 2D camera for individual and efficient performance – ideally that the algorithms are already integrated into the image sensor.

This paper is organized as follows: the next section describes essential notation of optical flow for motion estimation including the transformation of ground plan to the image plan for our approach and block matching algorithm. Section 3 presents the hardware platform of the vision system of Khepera mini-robot. The implementation of our optical flow algorithm for obstacle detection and the efficient resource utilization are discussed in section 4. The last section presents the paper conclusion.

2 Optical Flow and Block Matching

The main objective of the optical flow technique is to determine correspondence pixels of moving objects in the image sequence. This technique refers to the optical flow constraint equation, which gives the information about the spatial movement of image pixel and the interval time.

2.1 Optical Flow Constraint Equations

Optical flow is defined as displacement of pixels in the serial images, which is corresponding to the projection of 3D motion of objects. The principle of optical flow is not occupied for object recognition or 3D reconstruction, but motion estimation of pixels on the image sequence. The most well known model for estimating optical flow is image intensity model. The model assumes the continuous image intensity to be function of position and time given by in a Taylor series [5]:

$$I(x+dx, y+dy, t+dt) = I(x,y,t) + \frac{\partial I}{\partial x}dx + \frac{\partial I}{\partial y}dy + \frac{\partial I}{\partial t}dt + \ldots \quad (1)$$

The high order terms usually ignored. Assume that the image at some time $t + dt$ is the same original image at time t being moved translationally by dx and dy, then $I(x + dx, y + dy, t + dt) = I(x, y, t)$ thus

$$-\frac{\partial I}{\partial t} = \frac{\partial I}{\partial x}\frac{dx}{dt} + \frac{\partial I}{\partial y}\frac{dy}{dt}$$

The velocity follow x and y axis are $\frac{dx}{dt} = v_x$, $\frac{dy}{dt} = v_y$ and $\vec{v} = (v_x, v_y)$. Thus the equation is expressed as: $\frac{\partial I}{\partial t} = -(\frac{\partial I}{\partial x}v_x + \frac{\partial I}{\partial y}v_y) = -\nabla I \cdot \vec{v}$

The constraint equation shows the relation between the change of pixel intensity in time domain and the velocity of the pixel intensity in spatial domain.

2.2 Optical Flow Vector for Obstacle Detection

The main purpose of our approach is to realize the autonomous obstacle avoidance system of Khepera by using a 2D vision module. The vision module is mounted to the top of the mini-robot, which moves straight on or turns right or left on a planar plane. The camera looks at the plane, with ϕ slant angle to the plane, and has constant distant form the plan, illustrated in Fig. 1.

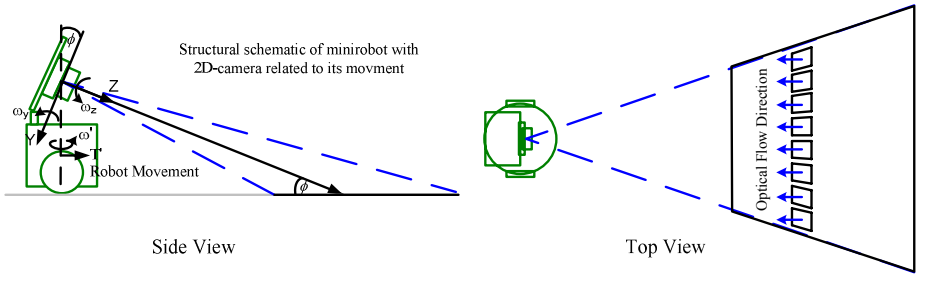


Fig. 1. The schematic illustrates robot's view and 8 detection blocks

Therefore the translation and rotation of the camera is constrained in $T_y=0$ and $\omega_y=0$. Thus, the image pixel velocity of X and Y components are derived from translational and angular velocity (T' and ω'), as shown in equation (2) and (3).

$$v_x = \frac{T_z x}{Z(x,y)} - \omega_y f + \frac{\omega_y x^2}{f} + \omega_z y = \frac{T'\cos(\phi)x}{Z(x,y)} + \omega'\cos(\phi)f + \frac{\omega'\cos(\phi)x^2}{f} - \omega'\sin(\phi)y \quad (2)$$

$$v_y = \frac{T_z y - T_y f}{Z(x,y)} - \frac{\omega_y xy}{f} - \omega_z x = \frac{T'(\cos(\phi)y + \sin(\phi)f)}{Z(x,y)} + \frac{\omega'\cos(\phi)xy}{f} + \omega'\sin(\phi)x \quad (3)$$

where $Z(x,y)$ is the distant between the camera lens and the planar plan ; f is the focus of camera lens. These constraint equations explain the relation between the velocity of the robot and the displacement of the corresponding pixels on the scene. Therefore,

when the speed of robot is defined, the optical flow fields and the processing time can be estimated. While the robot goes straight on without any obstacle the robot, the optical flow vectors and its constraint values are steady and near constant. When the robot approaches to an object obstacle, the optical flow vector will be rapidly longer than before. In contrast with drop-offs obstacle, the optical flow vector will be shorter. Therefore the mini-robot will be able to detect any obstacles, and then turn away to avoid hitting with the obstacles.

2.3 Block Matching for Correspondence

In digital image world, there are many simple standard techniques for evaluation of the optical flow vector and the correspondence blocks in the sequence images, for example using block matching and matching criteria, such as SAD, SSD and cross correlation [6]. Size and location of the blocks are usually defined depending on the appropriate applications. Each block from the previous image is matched into a block in the current image by shifting the previous block over the searching area of pixels in current image as shown in Fig. 2.

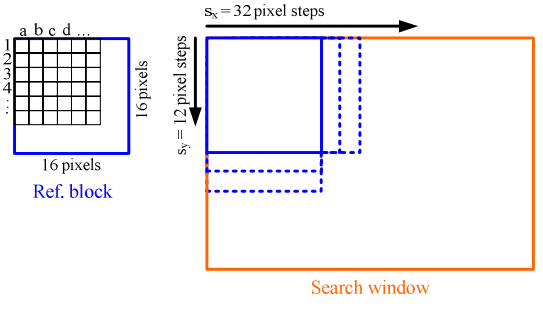


Fig. 2. Illustration of block matching by matching a reference block for every position through a searching area

At each shift, the sum of comparing parameter e.g. intensity or color of the two blocks is computed and saved. The sum parameter is called "match strength". The shift which gives a best result of the matching criteria is considered as the best match. The following equations are various criterions of block matching estimation, which are often implemented to determine the best match.

- Sum absolute difference (SAD) equation

$$SAD = \sum_{j=0}^{n} \sum_{i=0}^{m} | A_{(x+i,y+j)} - B_{((x+s_x)+i,(y+s_y)+j)} | \tag{4}$$

- Sum square difference (SSD) equation

$$SSD = \sum_{j=0}^{n} \sum_{i=0}^{m} (A_{(x+i,y+j)} - B_{((x+s_x)+i,(y+s_y)+j)})^2 \tag{5}$$

- Cross correlation equation

$$CORR = \sum_{j=0}^{n} \sum_{i=0}^{m} (A_{(x+i,y+j)} * B_{((x+s_x)+i,(y+s_y)+j)}) \tag{6}$$

where: $A_{(x,y)}$, $B_{(x,y)}$ are the intensity at (x,y) of images
m, n are the width and height of the reference block
s_x, s_y are pixel-shift steps in the x- and y-direction

The best match of SAD and SSD algorithms causes the least difference of the matching blocks that results the minimum SAD and SSD value. While the maximum value of cross correlation is considered as the best match

3 Hardware Platform for Implementation

The mobile robot for implementing obstacle avoidance is Khepera, which is widely used for research and edutainment. The robot's data processing part consists of a Motorola 68331 microcontroller, running with 25MHz clock frequency, 512 kB RAM and 512 kB. It has two independently driven wheels with incremental encoders for the odometry. Khepara provides 8 integrated IR sensors for closed distance and ambient-light measurement. The addition battery module supplies extension energy to the mobile robot for the long-run experiments. Khepera is equipped with the additional turret (illustrated in Fig. 3), for example, the camera module for capturing image and the FPGA module for processing real-time applications such as the image processing.

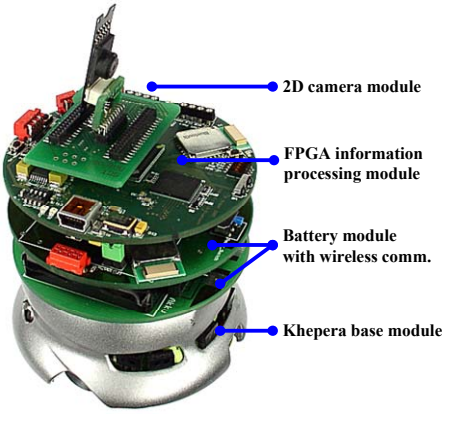


Fig. 3. The mini-robot Khepara equipped with the 2D color camera and the universal FPGA – MCU module for vision application

2D camera module is a miniature 2D color CMOS image sensor TC5740MB24B from TransChip Inc., in which color processing and JPEG codec are integrated. The camera can provide VGA resolution color image (max. 648×488 pixel), 10 bit/pixel and up to 40 fps (frames per second) at QVGA format. The integrated image

processor enhances the flexibility of the camera for image format (e.g. RAW, YUV, RGB) and automat setting (e.g. exposure control, white balance and digital zoom function). The power consumption of the camera is low about 85 mW at 2.8 V.

The robot's base module is able to operate the common processing tasks and motor controlling for navigation. Unfortunately the processing performance of the robot's base module isn't sufficient for our approach including 2D image acquisition and image processing. To solve this performance problem, we have designed an Universal FPGA–Microcontroller (MCU) Module [7] for the mobile robot. This module extends the processing performance and provides the communication option for robot to host and Ad-hoc communication in cooperative robotics. FPGA provides flexibility of programmable logic circuit to accelerate performance for many real-time critical applications and customized approaches. Thus the FPGA is often implemented for parallel computing and pre-processing sensor's data by coupling nearby a host processor and sensors (e.g. 2D image camera, A/D converter of signal receiver).

The universal FPGA-MCU module (illustrated in Fig. 4) consists of the main following components: Xilinx FPGA Virtex-E XCV300E-6, Cypress USB 2.0 MCU, SRAM & flash memory, ADC, DAC and Bluetooth module. The FPGA possesses 9612 logic blocks, 16kB block RAM and flexible I/Os, will be responsible for image acquisition and concurrently processing of image data to transmit symbolic data to the mini-robot's base. The FPGA can be reconfigured to change its tasks during run-time and also supports partial reconfiguration. The Cypress EZ-USB FX2 microcontroller provides control functionalities, e.g. FPGA reconfiguration, downloading of FPGA configuration file and fast transferring data such as image, lived-stream video.

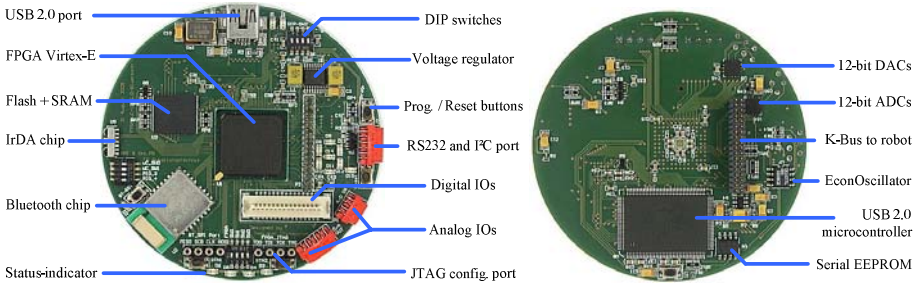


Fig. 4. The top and bottom view of the universal FPGA – MCU module for the mini robot

In the FPGA-MCU module, there are 128MBit Flash and 32 MBit SRAM, which is able to store up to 32 full bit-stream files for FPGA configuration. The Dual 12 bits channel ADC and Dual 12 bits channel DAC are the fast converters with ultra low power consumption using for sound applications, analog signal acquisition applications and analog control system (e.g. the back-light of LCD). The Mitsumi Bluetooth module contains BlueCore2-External [8] and a 16bit RISC microcontroller supporting for mobile applications of wireless communication such as Pico-nets and point-to-point connections.

4 Implementation of Optical Flow on Khepera

4.1 Implementing Model of Optical Flow

To detect obstacles using optical flow technique, the block matching SAD algorithm is implemented in the Khepera's vision module. Beginning with the camera provides raw RGB mosaic image sized 640 x 480 pixels to the on-board FPGA. The complete functionalities of camera control, data acquisition and image processing algorithm are realized in FPGA using hardware definition language (VHDL). The implementation cannot deal with the entire consecutive images due to memory space constrain of the block RAM memory inside FPGA, therefore the non-evaluated part of image will be ignored. The image acquisition part is responsible for carrying out the following functions: sub-sampling image pixels, cutting non-relevant area and saving the data into internal dual port synchronous RAM. Thus, the size-reduced image block is now in grey-scale, sized 256 x 24 pixels cutting from middle area of the raw RGB image. In this image block, there are 8 sub-blocks sized 30x16 pixels defined as the reference blocks for SAD operation and the obstacle detection area as well (illustrated in Fig. 5). The split into 8 sub-blocks can reduce the complex processing of enormous data. The SAD can be processed immediately after finish writing image into the FPGA.

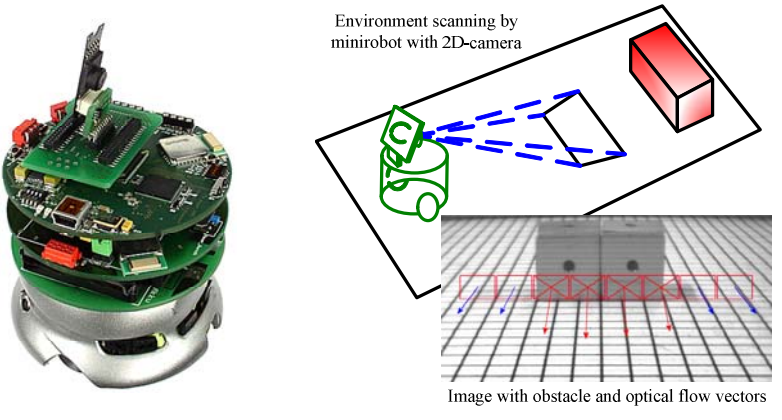


Fig. 5. The robot view with the 8 detection sub-blocks to process the optical flow fields

The optical flow vector for the blocks matching is between (0, 0) and (16, 8) pixel in X-Y image plane. The previous and current images are hold separately into two swap areas in the dual port block RAM of FPGA, liked as double buffering technique, which can provide the highest performance for simultaneously access of both images to process the SAD algorithm (shown in Fig. 6).

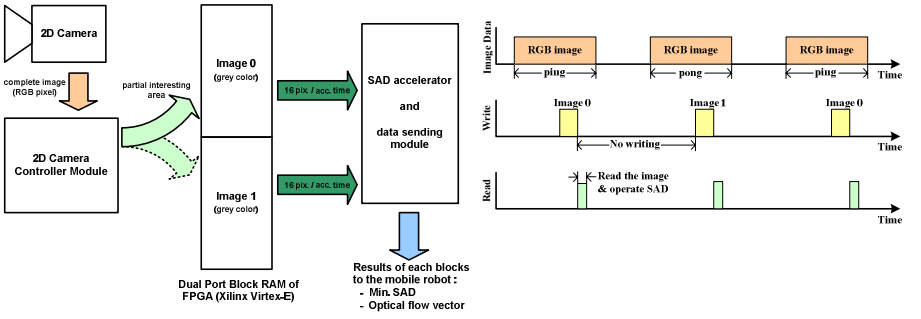


Fig. 6. The block diagram and time diagram of image processing model for the image acquisition and the SAD processing

4.2 Realization of SAD

To achieve the real-time requirement, effective operation and robustness for dealing with the enormous image data, the SAD of the block matching algorithm is preferred as the best choice. The SAD operation involves commonly a subtraction, taking the absolute value of the subtraction, and then accumulating the result, that is expressed in equation (7).

$$SAD_{(16\,pixels)} = \sum_{i=0}^{15} | A_{(x+i,y+j)} - B_{((x+s_x)+i,(y+s_y)+j)} | \tag{7}$$

Where (x, y) is reference pixel position in the image ($0 \le x; y \le$ image size-16).
$A(x;y)$ is pixel intensity in the previous image at $(x; y)$.
$B(x;y)$ is pixel intensity in the current image at $(x; y)$.
$(s_x; s_y)$ is the searching vector.

The pixel intensity for the computer vision is commonly represented in 8 bit discrete signal as (256 levels grey-scale). Thus, data A and B are declared to be unsigned 8 bit numbers. The 1x16 SAD equation is explained that the absolute differences ($|A - B|$) of 16 pairs of pixels can be evaluated simultaneously, and then being accumulated in next step [9]. This parallel executing method reduces the operating time of iterations and increases the effective computing performance. Considering the smallest function of SAD, the Subtraction of two unsigned numbers $(A - B)$ can be derived by adding A with a bit inverted B ($B' = 2^n - 1 - B$) and a compensator one, where $A \ge B$.

$$A - B \;=\; A + (2^n - 1 - B) + 1 \;=\; (2^n + A - B)_{8bit} \tag{8}$$

The 2^n of last term can be ignored (i.e. $2^8 = 100000000b$), because we considerate only 8 bit data.

To determine the smaller operand between A and B, this is applied by inverting one of the operands, and calculating the carry-out, whether the carry-out would be generated by addition of the both operands. For example: assume $B > A$, consider the following inequality:

$$\begin{aligned} B - A &> 0 \\ B + 2^n - 1 - A &> 2^n - 1 \\ B + A' &> 2^n - 1 \\ A' + B &\geq 2^n \end{aligned}$$

This means the carry-out is always generated where $B > A$. In contrast, the carry-out isn't generated where $A \geq B$. Thus, the smaller operand is inverted by EX-RO with one corresponding to carry-out or inverted carry-out of the carry generator. The bigger operand keeps the same value. The inverted smaller and bigger operands are sent to the adder tree in next step. The schematic logic of the operand preparation and inversion of the smaller operand between *A* and *B* can be illustrated in Fig. 7a.

The absolute differences of the two operands *(A, B)* can be evaluated from the sum of the bigger operand, the inverted smaller operand and compensator one. In our implementation, the determination and inversion of smaller operand are done in parallel operations of 16 column-pixels in the previous and current images. Therefore, there are 32 operands for all 16 absolute operations and the accumulated compensator is 16. The adder tree of 32 operands can be designed in several types. In this approach, a tree of Carry-Save-Adders (CSA) is implemented in 8 levels using 240 carry save adders. The outputs of adder tree are 8bit carry term and 12 bit sum term. To evaluate the 1x16 SAD, the both terms are given to 8-bit carry look-ahead adder, whose carry bit is set to 1 derived from the compensator valued 16. The last result of 1x16 SAD is 12bit binary number. The structure of this 1x16 SAD computation can be illustrated in Fig. 7b.

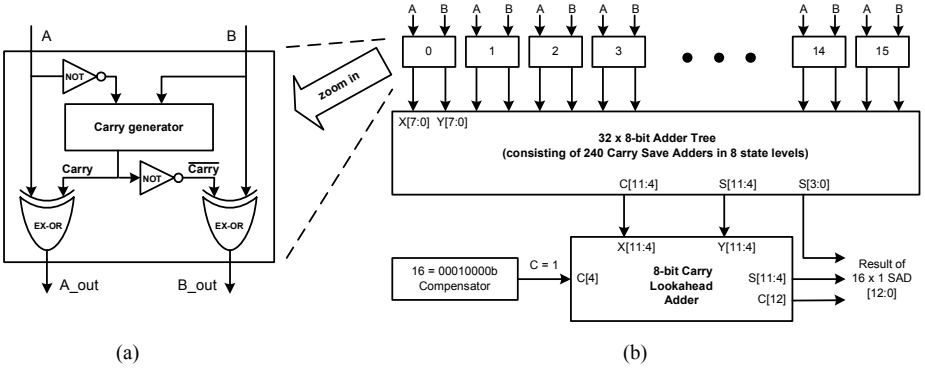


Fig. 7. (a) The logic diagram of operand preparation by using carry generator to consider the bigger value of 2 operands (A, B); (b) The block diagram of 1x16 SAD operations in parallel computation using 16 modules of operand pre-processing, an adder tree with 32 inputs and an 8bit carry look-ahead adder

Our implementation acquires the 32 pixels in one access time by using the internal dual–port block RAM in FPGA, which has an essential role for the performance. Our design provides 24 banks of 512 byte for write/read 24 pixel rows. The first 256 byte is used for *(A)* block and the last 256 byte for *B* block. Each buffer block is updated

alternately with the current image, so the SAD design can effectively utilize the high performance and parallelism of the dual-port synchronize block RAM.

To complete the 30x16 SAD of block matching, the 1x16 SAD operation has to be repeated 30 times by computing the next column pixel. The result of each operation is accumulated to evaluate the SAD value at its own position. The block matching is then repeated through all position in the searching area from (0,0) to (16,8), that means 153 times of block matching are operated to determine the best match of each reference sub-block. The best match position belongs to the position given the minimum 30x16 SAD value. Next, the similar processes will be performed for the next sub-blocks until complete all 8 sub-blocks are completed. The flow vector and its SAD constrained value of all sub-blocks are sent to mobile robot platform. The robot evaluates any possible obstacles and controls the wheels to avoid hitting the obstacle.

4.3 Result and Resource Utilization for the Implementation

The implemented modules by using VHDL, i.e. the 2D camera controller, the image acquisition, the blocks memory controller and the SAD algorithm, are completed and perform all designed functions. These modules are applied into the FPGA Virtex-E XCV300E running at 48 MHz. The 30x16 SAD operation of the 8 sub-blocks can be calculated within 148000 clock cycles or 3.08 ms, which is 16 times faster compared to a typical processor solution at the same clock rate, due to the processor can only access each pixel per operation. The image acquisition from the camera can reach up to 12 frames per second, while the SAD operation of the 8 sub-blocks can be run faster over 300 frames per second. The problem is the bottleneck of image data acquisition from camera sensor to FPGA, which can delivery the image by each pixel. Thus, the FPGA module can be idle in sleep mode (as option) after finishing the SAD process to save the energy consumption.

After the place and route of the VHDL code of the design on the Virtex-E FPGA, over 60% of the available slices are available for other image processing algorithms (see table 1). Power consumption of the FPGA for camera controller and optical flow computation is about 125 mA estimated by using XPower in Xilinx ISE software suite. The resource usage for the implemented algorithms is independent from the size of the raw RGB image. For utilizing another camera a few change of camera controller may be required.

Table 1. Resource Utilization on Xilinx XCV300E FPGA

Resource Utilization	Used	Available	Utilization
Number of Slice Flip Flops	411	6144	6%
Number of 4 input LUTs	2097	6144	34%
Number of occupied Slices	1167	3072	37%
Number of Block RAM	24	32	75%

5 Conclusion

The additional processing module based on FPGA platform provides flexible ability and enhances processing performance for the mini-robot. Therefore the mobile robot equipped with the FPGA module can be applied in various application areas and complex experiments. An obvious advantage of the FPGA coupling to image sensor is that the complex processing tasks are customized by a suited algorithm and then implemented in the FPGA to deal with the enormous image data. Consequently, the data bandwidth and data transaction between image sensors and robot's base module are minimized. The main processor is relieved from complex tasks. In our application scenario the robot equipped with a 2D vision module is able to detect obstacles in its environment for navigation. The image acquisition and the image processing aren't computed by the main processor of the robot, but by the additional FPGA.

The entire processing steps, from image acquisition, optical flow computation until result transaction to mini-robot base module, are implemented in the Virtex-E FPGA (XCV300E) by using VHDL. The implementation requires the 37% of available slices, and 75%of block RAM from the FPGA, that means this approach can also implemented in other FPGAs providing less slices and hardware resource. The on-board FPGA can be utilized in its remaining resources for further image processing algorithms or other computation tasks, as well. In the optical flow approach, the image processing module operates at 48 MHz and processes 12 images per second, where 8 image sub-blocks sized 30x16 pixels are used for the optical flow analysis. During exploration on planar plan, Khepera robot captures the sequence images and calculates the flow vectors in real time. As soon as Khepera detects an obstacle in the front of it, the mobile robot will turn away to avoid the collision.

Acknowledgement

This work was partly supported by the Sixth Framework Program of the European Union as part of the GUARDIANS project (no. 045269, www.guardians-project.eu).

References

1. Chinapirom, T., Kaulmann, T., Witkowski, U., Rückert, U.: Visual object recognition by 2D-color camera and on-board information processing for minirobots. In: Proceeding of the FIRA Robot World Congress, Busan, Korea (2004)
2. Franceschini, N., Pichon, J.M., Blanes, C.: From insect vision to robot vision. In: Philosophical Transactions Royal Society of London, vol. 337, pp. 283–294
3. Beauchemin, S.S., Barron, J.L.: The computation of optical flow. ACM Computing Surveys 27(3), 433–466 (1995)
4. Kleist, T.: Bee navigation: the eyes have it - photoreceptors in bee's eyes used to navigate. In: Science News (October 4, 1986)
5. Ballard, D.H., Brown, C.M.: Optical flow. In: Computer Vision, pp. 102–105. Prentice-Hall, Englewood Cliffs (1982)

6. Trucco, E., Verri, A.: The correspondence problem. In: Introductory Techniques for 3-D Computer Vision, pp. 145–150. Prentice Hall, Inc., Englewood Cliffs (1998)
7. Chinapirom, T., Witkowski, U., Rückert, U.: Universal FPGA Microcontroller Module for Autonomous Minirobots. In: Proceeding of the Autonomous Minirobots for Research and Edutainment (AMiRA 2005), Fukui, Japan (September 2005)
8. Cambridge Silicon Radio (CSR): BlueCore2-External. In: BlueCore2-External Product Data Book (2004)
9. Vassiliadis, S., Hakkennes, E.A., Wong, J.S.S.M., Pechanek, G.G.: The Sum Absolute Difference Motion Estimation Accelerator. In: 24th EUROMICRO Conference (EUROMICRO 1998), vol. 2, p. 20559 (1998)

Ad-Hoc Communication and Localization System for Mobile Robots

Bernd Neuwinger[1], Ulf Witkowski[2], and Ulrich Rückert[1]

[1] System and Circuit Technology, Heinz Nixdorf Institute, University of Paderborn, Paderborn, Germany
berndn@hni.upb.de
[2] Electronics and Circuit Technology, South Westphalia University of Applied Sciences, Soest, Germany
witkowski@fh-swf.de

Abstract. Robots exploring unknown territory cannot rely on fixed infrastructure for communication and localization purposes. Therefore an ad-hoc communication and localization system is required to provide such facilities. In this paper a solution for a radio based localization system is presented using a 2.4 GHz wireless radio transceiver based on Chirp Spread Spectrum (CSS) modulation, which allows distance measurements by evaluating the signal propagation delay. Using odometry data as additional information source a communication and positioning system is developed fulfilling ad-hoc demands.

1 Introduction

Robots exploring unknown territory cannot rely on fixed communication and localization infrastructure. Thus, an ad-hoc communication and localization system is required allowing the robots to communicate among each other and locate themselves. A typical example scenario is given by the GUARDIANS project [1]. The objective of this project is to assist firefighters in the matters of orientation and communication during the exploration of a burning target building. For archiving this goal a radio frequency based positioning and communication system is considered to be suitable since smoke and other obstacles can affect the optical (laser and vision) perception of the robots during a fire. In this paper the main focus will be kept on the ad-hoc localization system.

Because no fixed infrastructure for determining the robot's position can be assumed within a target building a positioning system should work autonomously, thus, provide position data relatively to a predefined reference point. This point could be for instance the entry of a target building. Therefore a relative locating system based on measured angles and distances would be desirable. This is shown in Fig. 1.

Unfortunately the determination of a radio signal's reception angle is on the one hand very complicated to realize in terms of antenna complexity and on the other hand sensitive to multipath scattering [2]. High timing requirements lead to complicated hardware structures, high costs and high energy consumption.

J.-H. Kim et al. (Eds.): FIRA 2009, LNCS 5744, pp. 220–229, 2009.

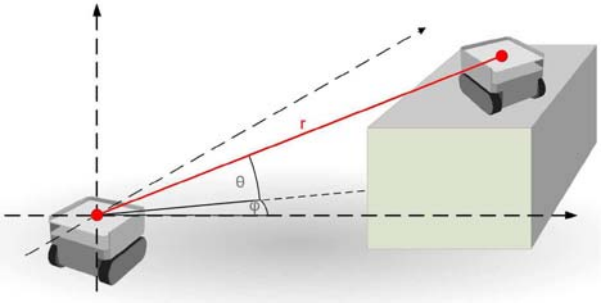


Fig. 1. Robot locating another robot based on angle and distance measurements

Hence for a stable locating system, which is capable to provide position data within a fixed timeframe, so called real-time locating system, other information than the reception angle is needed to calculate a position. A promising solution could be trilateration. This method determines the position of a robot (tag robot) based on distance measurements to other points, in this case other robots (reference robots). But the coordinates of these reference robots have to be known for the calculation. A solution to this could offer the use of odometry. For instance a few robots could drive guided by odometry from a common start point to certain positions thereby providing an infrastructure for the tag robot to determine its position. Figure 2 shows a possible setup of such a system.

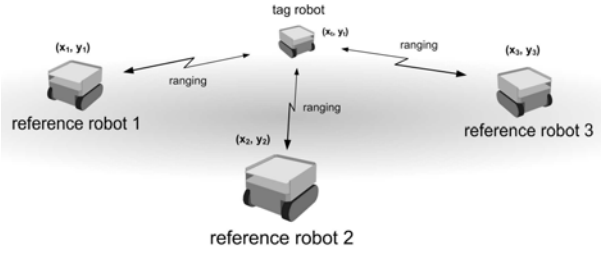


Fig. 2. Robot determining its position through trilateration

The technical concept of all robots in Fig. 2 is the same, the only difference between them is the knowledge of their position. Besides the position data of the reference robot distances between the robots have to be measured. Using radio frequency signals a Time-Of-Flight (ToF) implementation takes advantage over a Receive Signal Strength Indication (RSSI) solution [3]. In the ToF-approach the roundtrip time from one robot to another and back is measured allowing

the calculation of the distance between both robots taking into account the propagation speed of electromagnetic signals.

Two different techniques are promising good results regarding distance measurements. These are Ultra-Wideband (UWB) and Chirp-Spread Spectrum (CSS). UWB is considered to reach accuracy in the scope of centimeter [4,5], but due to the fact that UWB hardware development is still in the beginning not many transceivers are available. CSS reaches a measurement accuracy of about 1 to 2 meters indoors [6]. This is worse than UWB, but there are already transceivers available supporting ToF-measurements, for instance the nanoLOC-Chip of Nanotron, which let this technique appear more attractive [6].

In the following a real-time location system is examined based on both the nanoLOC-Chip of Nanotron and odometry data of the robots equipped with the communication device. Besides ToF-measurement capabilities the nanoLOC provides wireless communication up to 2 Mbits. A prototype system is implemented using the BeBot robot of the University of Paderborn [7] and nanoLOC Development Kit [8]. For keeping the system simple it will be limited to the use in a two dimensional plane. An extension to the third dimension is possible.

This paper is structured as follows: In Sect. 2 the measurement behavior of the used nanoLOC-Chip will be examined. Algorithms for calculating a position out of the measured raw data will be presented in Sect. 3. Then in the subsequent section field measurement will be made for analysing the system accuracy. Thereafter in Sect. 5 a prototype system will be shown and, finally, in the end of this paper a conclusion will be given.

2 Examination of NanoLOC's Distance Measurement Behavior

Before implementing an ad-hoc real-time locating system the measurement behavior of the nanoLOC-Chip has to be investigated. Therefore different distance measurements were made using the Nanotron's nanoLOC Development Kit. In all of these measurements the distance between two nodes is gradually increased keeping one node at a fixed position and moving the other one. At first a series of outdoor measurements were recorded. The Fresnel zone is disregarded, since it does not represent a typical application scenario. Figure 3 shows the absolute measurement error dependent on the distance for this measuring.

It can be seen that the absolute measurement error grows mainly linearly up to the range of 27.5 m, further on the behavior is not deterministic anymore. For the regression line the values above 27.5 m are disregarded. The non-deterministic behaviour is probably due to a bad Signal-to-Noise Ratio (SNR) resulting in a misdetection of CSS pulses. Further measurements confirm the approximately linear interrelation.

Then indoor measurements were performed. Figure 4 shows an indoor distance measurement under Line-Of-Sight (LOS) condition in a large, furnished room.

Based on this measuring an accurate approximation cannot be found, but for practical issues a linear approximation works out well. Figure 5 shows an indoor

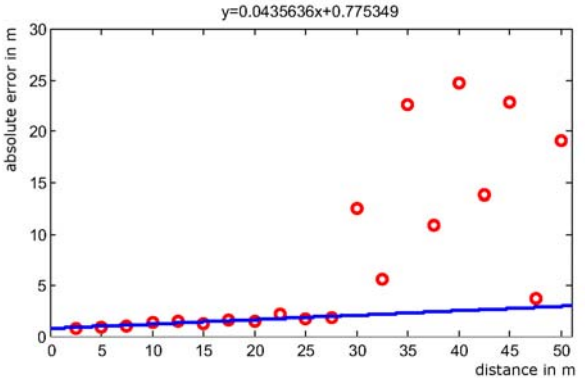


Fig. 3. Outdoor measurement: absolute error dependent on the distance. Distance resolution: 2.5 m.

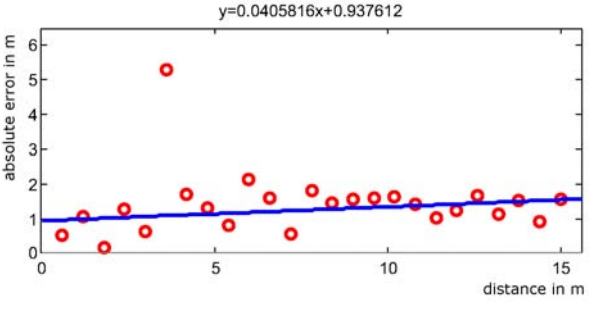


Fig. 4. Indoor LOS-measurement: absolute error dependent on the distance. Distance resolution: 0.6 m.

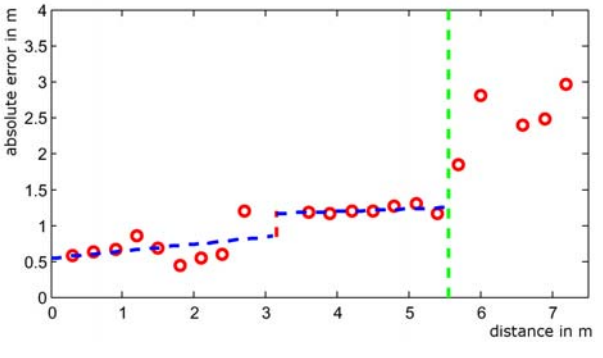


Fig. 5. Indoor NLOS-measurement: absolute error dependent on the distance. Distance resolution: 0.3 m.

distance measurement under mixed conditions, Line-Of-Sight (LOS) and Non-Line-Of-Sight (NLOS) condition. The two nodes are placed in the same room for distances below and in adjacent rooms for distances above 3 m.

As already noticed in the outdoor measurement the behavior in the outer region (5.5 m and above, indicated by the green dashed line) exposes none deterministically and thus this values are disregarded for the regression lines. The first part of the graph can be interpreted as a set of two linear functions with an offset at a distance of about 3 m, where the wall is inhibiting the direct Line-Of-Sight (LOS). This can be explained with the different speed of electromagnetic waves in different materials.

In all examined scenarios, outdoor, indoor LOS and indoor NLOS, an approximately linear error behavior can be found. Due to a lack of precise information on how the nanoLOC-Chip works, it was not possible to find out a satisfying explanation for this behavior. But considering practical issues the result enables to improve the calculation of a positioning system.

3 Algorithm for Calculating a Position

For determining a two-dimensional position in a plane at least three reference robots are needed. In general n+1 reference points are prerequisite for the calculation of n-dimensional coordinates. Besides the number of reference points there are also demands to their arrangement. The points must be spread over the complete n-dimensional space. If the arrangement only fills an m-dimensional subspace, multiple solutions will be found in an (n-m)-dimensional subspace.

Using three or more reference robots, which will get their position information through odometry, a real-time locating system based on the nanoLOC-Chip can be set up. Therefore methods for calculating a position out of collected raw data are required. Two promising methods are the least square method and the Cayley-Menger-determinant method. The least square method is a numeric procedure, which calculates the point that minimizes the distance deviation of all measurements. For implementation the Gauss-Newton algorithm can be used [9]. In contrast the Cayley-Menger-determinant method is an analytic procedure, which calculates a point through geometric interrelations [10]. Since this method is not numeric approximation like the least square method, it promises a lower computing time.

But before calculating the final tag robot position the algorithm should improve the raw data in order to make the results more stable and reliable. This will be realized in two steps. In step one the systematic error, which was examined in Sect. 2, will be removed. Since this error can be approximated by a linear function the calculation effort is very low:

$$r_{corrected} = \frac{r_{measured} - b}{1 + m} \ , \tag{1}$$

with m as slope and b as y-intercept.

In step two unreliable measurements will be removed or adapted by analyzing the raw data. Theoretically the measurements results in three or more circles intersecting in one point. Due to the inaccuracy of the nanoLOC system the circles will not intersect exactly at one point, considering an error of 1 m or more for each measurement the intersections can diverge significantly from the theoretical value. This leads to two problems. On the one hand the algorithm must be robust against strongly varying values, and on the other hand misdetections are hardly distinguishable from correct measurements. Therefore a solution must be found to judge the quality of the measurements.

Analyzing the cutting angles of the circles is not sensible, since the inaccuracy of the nanoLOC system derives in a strong deviation of angles [11]. Instead conditions were examined whether circles intersect or include each other. Involving three or four reference robots 11 respectively 57 non redundant cases could be found. Figure 6 shows exemplarily all cases that can appear for three reference robots.

Assuming that the reference points are arranged in almost triangle respectively rectangle shape predictions about the reliability of the measurement can be made. Figure 7 shows an example of case no. 5.

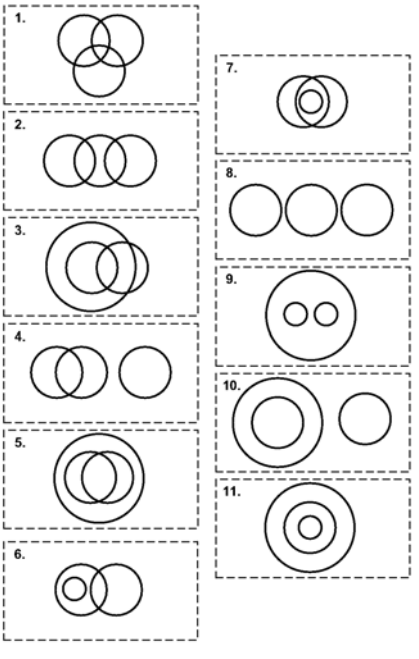


Fig. 6. Non redundant intersect/include cases for three reference robots

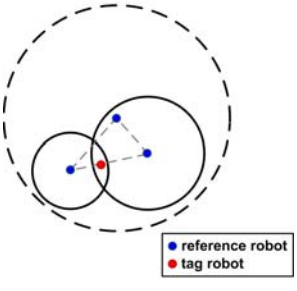


Fig. 7. Example case for 3 reference robots and triangle arrangement

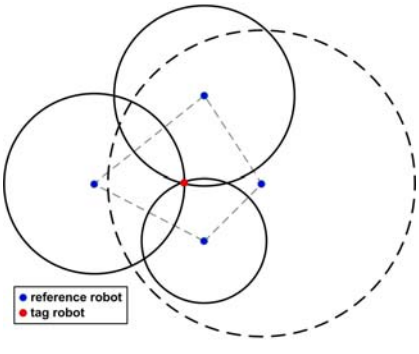


Fig. 8. Example case for 4 reference robots and rectangle arrangement

Since the dashed circle is including the other circles, it is very likely that this measurement failed. In this case the radius of this circle can either be adapted, for instance being reduced until all circles intersect, or be used to eliminate multiple solutions in the calculating methods. Alternatively a measurement could be declared as invalid if the combination of circles did not promise a reliable solution. Figure 8 shows an example using four reference robots. In this case the dashed circle can be neglected.

4 Field Measurements

For testing the algorithms and determining the accuracy of the complete system field measurements were performed. Therefore rectangular fields in different environments were divided by a grid of 60 cm and for each point of these grids the position was determined by the nanoLOC system and the algorithms of Sect. 3. The reference nodes were placed in a rectangular respectively triangular arrangement within the measurement site. All results were averaged over 100 measurements and compared to the real position value. To analyze the influence of modifications of algorithm different calculation and error correction methods, which have been discussed in Sect. 3, were used for determining the position.

Table 1 shows the results of a field measurement within a big, unfurnished room. The field has a dimension of 11.4 m x 11.4 m and the values were calculated with the use of systematic error removal and raw data correction. These results are in detail the average, the median as well as the upper and the lower decil of the absolute error (difference between measured and real position) of all points within this measurement site.

As it can be seen from the table the Cayley-Menger-determinant method provides worse results than the Least Square method. But since the differences are not significant and the Least Square algorithm needs a lot more computing time the Cayley-Menger-determinant method is preferable for an operation on a mobile device with limited resources like a robot. A combination of both methods as discussed in [11] turned out not to be advantageous, because the criteria used there are not applicable for the nanoLOC system. As expected the use of four reference robots instead of the least necessary three reduces the error of the calculated position. Figure 9 shows the measurement field without the use of systematic error removal and raw data correction.

Table 1. Results for test field measurement: absolute error in m

	Cayley-Menger 3 robots	Least Square 3 robots	Cayley-Menger 4 robots	Least Square 4 robots
average	1.46	1.43	1.00	0.96
median	0.70	0.68	0.64	0.54
upper decil	0.20	0.18	0.16	0.15
lower decil	3.12	2.99	2.08	2.00

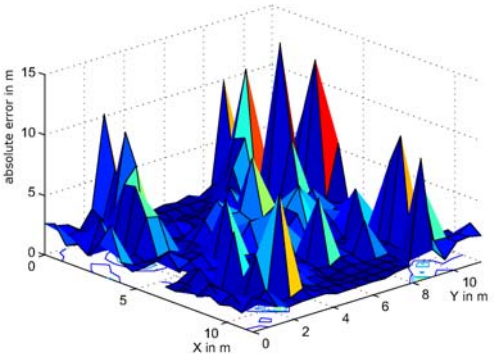


Fig. 9. Result graph for test field without using systematic error removal and raw data correction

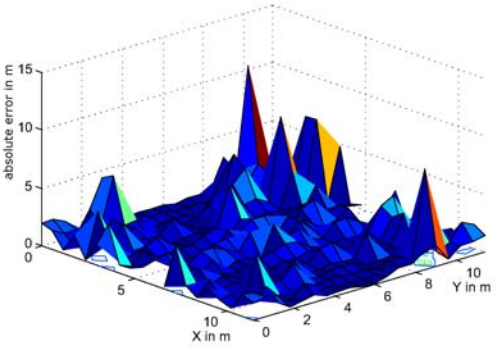


Fig. 10. Result graph for test field using systematic error removal and raw data correction

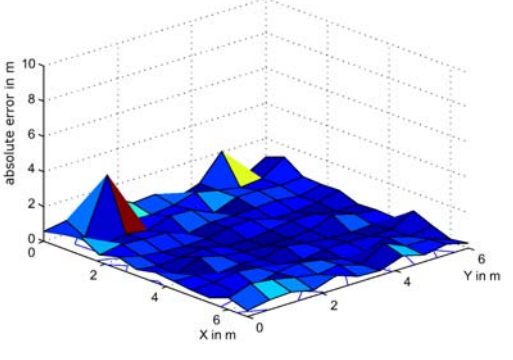


Fig. 11. Result graph for a small test field using systematic error removal and raw data correction

In Fig. 10 the same measurement field can be seen using systematic error removal and raw data correction. Comparing both graphs two major differences are important. On the one hand in Fig. 10 the whole error level lowered in comparison to Fig. 9. On the other hand fewer spikes appear in Fig. 10. These affects can be explained through the improvement in the algorithm. The systematic error removal lowers the average error and thus the error level. The raw data correction filters wrong detected values and consequently prevents spikes in the graph. Another aspect regarding both graphs is that the highest errors occur at the outer regions of the test field. This might be explained by the limited range of the nanoLOC chip as visible in Fig. 3 and Fig. 5. A measurement on a smaller test field (on a different site), presented in Fig. 11, shows better results. At this test site an average error of 45 cm could be reached.

5 Prototype System

By connecting the nanoLOC development boards included in the nanoLOC Development Kit to the robot platform BeBot [7], which was developed at the University of Paderborn, a first prototype system was implemented. All robots, in total five, are constructed the same way so that each robot can either be a tag or reference robot. A neighborhood algorithm was implemented enabling the robots to know which other robots are within reach and whether they are able to provide a position value. On a PC, which is connected to the system via a USB communication stick, the position of the tag robot was visualized.

Figure 12 shows a BeBot extended by a nanoLOC Development Board. A sample setup of the location system can be seen in Fig. 13.

Fig. 12. Prototype: BeBot combined with nanoLOC Development Board

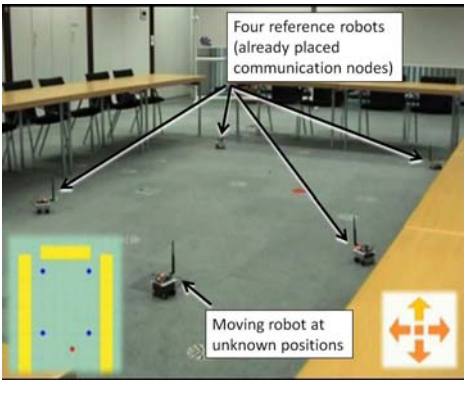


Fig. 13. Sample setup of nanoLOC based location system

6 Conclusion

Since the development of a location system based on signals' reception angle requires too much technical effort in mobile robot application an ad-hoc communication and localization system based on trilateration and odometry was implemented. Within a short range of about 5 - 10 m indoor the system provides acceptable results of about 1 m accuracy.

To improve the range of the system under LOS and NLOS indoor conditions a BeBot module board is currently under development, consisting of a nanoLOC-Chip (0 dBm output) combined with a 20 dBm amplifier. Further tests will show whether this module will affect range and reliability of the localization system as expected. Since more than five modules will be produced, tests with more than four reference points can be performed. In addition Kalman filtering in combination with use of odometry data will be implemented to improve the behavior of the system during the movement of robots [11]. In a further step the system could be extended to the third dimension.

Acknowledgments. This work was supported by the Sixth Framework Program of the European Union as part of the GUARDIANS project (no.045269, www.guardians-project.eu).

References

1. GUARDIANS Project, `http://www.guardians-project.eu/`
2. Roberts, R.: Ranging Subcommittee Final Report. Harris Corporation (2004)
3. Nanotron Technologies GmbH: Real Time Location Systems (RTLS), `http://www.nanotron.de`
4. Tüchler, M., Schwarz, V., Huber, A.: Accuracy of an UWB localization system based on a CMOS Chip. In: Proceedings of the 2nd workshop on positioning, navigation and Communication (WPNC 2005) & 1st ultra-wideband expert talk (UET 2005). Hannoversche Beiträge zur Nachrichtentechnik, pp. 211–220. Shaker, Hannover (2005)
5. Tüchler, M., Schwarz, V., Huber, A.: Location accuracy of an UWB localization system in a multi-path environment. In: IEEE International Conference on Ultra-Wideband, pp. 414–419, Zürich (2005)
6. Nanotron Technologies GmbH: nanoLOC TRX Transceiver Factsheet, `http://www.nanotron.de`
7. Witkowski, U., Herbrechtsmeier, S., El Habbal, M., Rückert, U.: Powerful Miniature Robot for Research and Education. In: IEEE Proceedings of the 5th International Conference on Computational Intelligence, Robotics and Autonomous System (CIRAS 2008), Linz (2008)
8. Nanotron Technologies GmbH: nanoLOC Development Kit User Guide V. 1.05, `http://www.nanotron.de`
9. Navidi, W., Murphy, W.S., Hereman, W.: Statistical methods in surveying by trilateration. Computational Statistics and Data Analysis 27(2), 209–227 (1998)
10. Thomas, F., Ros, L.: Revisiting Trilateration for Robot Localization. IEEE Transactions on Robotics 21(1), 93–101 (2005)
11. Lee, K.-W., Park, J.-B., Lee, B.-H.: Dynamic localization with hybrid trilateration for mobile robots in intelligent space. Intelligent Service Robotics 1(3), 221–235 (2008)

Robotic Implementation of Realistic Reaching Motion Using a Sliding Mode/Operational Space Controller

Adam Spiers[1,2], Guido Herrmann[2], Chris Melhuish[1],
Tony Pipe[1], and Alexander Lenz[1]

[1]Bristol Robotics Laboratory, Coldharbour Lane,
Bristol Business Park, Bristol
Avon, United Kingdom
BS16 1QD
[2] Department of Mechanical Engineering, University of Bristol,
Queen's Building, University Walk, Bristol,
Avon, United Kingdom
BS8 1TR
{A.Spiers,G.Herrmann}@bris.ac.uk
http://www.brl.ac.uk

Abstract. It has been shown that a task-level controller with minimal-effort posture control produces human-like motion in simulation. This control approach is based on the dynamic model of a human skeletal system superimposed with realistic muscle like actuators whose effort is minimised. In practical application, there is often a degree of error between the dynamic model of a system used for controller derivation and the actual dynamics of the system. We present a practical application of the task-level control framework with simplified posture control in order to produce life-like and compliant reaching motions for a redundant task. The addition of a sliding mode controller improves performance of the physical robot by compensating for unknown parametric and dynamic disturbances without compromising the human-like posture.

Keywords: Robotics, Sliding Mode, Operational Space, Human Motion, Parametric Uncertainty, Friction.

1 Introduction

Control of humanoid robot motion remains a challenging area of research. In particular high degrees of freedom (dof) means that almost every desired end effector motion generates an infinite number of motion possibilities and trajectories. In many cases humanoid robots are controlled in the same way as their industrial counterparts, with inverse kinematics and high-gain joint level control. This is suitable for the static industrial environment, where humans are separated from robots and tasks are repetitive, fast and precise.

J.-H. Kim et al. (Eds.): FIRA 2009, LNCS 5744, pp. 230–238, 2009.

For social robotics the working conditions are arguably more challenging with varying workspaces, payloads and a proximity to humans that imposes an important safety criterion. In such cases it is beneficial to have a more flexible control scheme that is able to solve general tasks with less stiff joints for safer human/robot physical interaction. Additionally, it has been suggested that if a humanoid robot's motion is predictable and human-like then human/robot interaction is more likely to succeed [1].

In this paper we present a robot controller that can replicate certain aspects of human motion while achieving practical levels of accuracy for real world application.

1.1 Human Motion

It is easily observable that the majority of human motion follows certain paths and constraints to produce generally consistent motion patterns [3,10]. The driving of motion by optimisation of a particular parameter has been popular in biomechanics with realistic motion being produced by various methods. Flash and Hogan [7] minimised jerk to model smooth motion of the hand in planar motion. Uno et al. [11] proposed that minimal joint torque change and muscle force change can be used as a driver for human motion.

De Sapio et al. [4] proposed optimisation of joint torques and minimisation of muscular effort in order to generate human motion trajectories over the entire body. In [9], the same group of authors described how multiple goals (i.e reaching for a target, maintaining balance and avoiding obstacles) may be executed in a hierarchical manner using decoupled controllers. Their method of achieving this is known as the *operational space* formulation in which the primary controller addresses the *primary* task of the motion to be executed (i.e. moving the end effector to a desired location) while the secondary *posture* controller addresses the system's redundant degrees of freedom. By assigning an optimal controller to the posture controller the authors are able to minimise 'effort', which produces human like motion.

2 Human/Robot Task Design

In this paper, we describe the application of the operational space formulation to control a two-dof robot for synthesis of human motion. Due to poor initial performance of the physical system (affected by sticktion/friction and model mismatch) we have created an additional sliding mode controller on the task level. This has improved performance of the robot.

An objective robot task was required that would allow evaluation of how *human-like* the robot's motion appeared to be. The task had to be an action that humans execute naturally and that could be executed by a robot with limited degrees of freedom (in order to reduce complexity at this early stage in the study) while also being scalable for more complex systems. It was also necessary to design a task that would induce nonlinear joint torques due to gravity (in order to produce a variable 'effort' measurement).

By choosing a vertical reaching motion, the objectives were fulfilled with the task being achievable with a 2dof robot. Ten human volunteers were instructed to touch or grab an object suspended above their head, in-line with the saggital plane of the shoulder (making the resulting trajectory approximately planar). Video analysis of these trials allowed prominent features to be observed in the reaching motion.

In order to make the reaching task suitable for verification of human motion synthesis an element of redundancy was added. By making the robot reach to a desired height we permit freedom in the horizontal plane. Hence, there is no unique kinematic solution to the task and more freedom is permitted in the reaching trajectories. This redundancy also provides an ideal foundation for the decoupling action of the operational space formulation, as the reaching task may now be separated into active (i.e. desired) and redundant elements. As in [4] we will use the secondary (posture) controller to drive redundant dof.

3 Controller Design

Design of the controller is based on a dynamic model of the robot system. This dynamic model takes the form of Equation (1).

$$A(q)\ddot{q} + b(q, \dot{q}) + g(q) = \Gamma, \tag{1}$$

Where Γ is a vector of system torques, q is the generalised (joint space) co-ordinate vector, A is the mass/inertia matrix, b are centrifugal/coriolis terms and g are gravity terms.

3.1 Task Control Framework

The task control framework is fully described in [9] and [4]. A short account of it is given here.

The task level controller functions by controlling a specific variable that represents the task to be undertaken. In the vertical reaching task described above, the task variable X_y is the cartesian height of the end effector. Thus, $X_y = X_y(q)$ is a function of the generalised co-ordinates, q, i.e., the joint angles.

By addressing only X_y, the redundant horizontal position of the end effector, X_x may be driven by the separate posture controller. This decoupling action is at the heart of the operational space formulation and permits us to define the hierarchical control method whereby achieving the target height takes priority over maintaining human posture.

The joint space system dynamics (1) are projected into the task space dynamics by joint variable transformations using the Jacobian $J = \frac{\partial X_y}{\partial q}$ so that the dynamics for $X_y(q)$ are $\Lambda(q)\ddot{X}_y + \mu(q, \dot{q}) + p(q) = f$, where:

$$\begin{gathered} \Lambda = (JA^{-1}J^T)^{-1}, \quad \mu = \bar{J}^T b - \Lambda \dot{J}\dot{q}, \quad p = \bar{J}^T g, \\ J = \frac{\partial X_y}{\partial q}, \quad \bar{J} = A^{-1}J^T(JA^{-1}J^T)^{-1} \end{gathered} \tag{2}$$

Assuming good parameter knowledge allows the system to be controlled by a feedback linearising controller and a PD controller $f^\star$ (3). X_{y0} is the desired constant height in the task space.

$$f = \hat{\Lambda}(q)f^\star + \hat{\mu}(q,\dot{q}) + \hat{p}(q), \quad f^\star = -K_x(X_y - X_{y0}) - K_v\dot{X}_y, \tag{3}$$

where $\hat{\Lambda}$, $\hat{\mu}$ and $\hat{p}$ are estimates of Λ, μ and p. Finally, using the Jacobian guarantees that the resulting torque, Γ_T for task level control, does not affect the posture controller: $\Gamma_T = J^T f$.

3.2 Simplified Posture Controller

De Sapio et al. [4] produce human like motion in simulation by minimising effort over a realistic number of antagonistic actuators with muscle-like properties. Using the capabilities of OpenSim [5], the authors are able to realistically simulate the changing actuator forces due to factors such as muscle orientation and the wrapping of muscle around bones. The dynamics of these actuators are captured in the activation matrix.

In order to simplify the Posture Controller of [4] we have reduced the complexity of the activation matrix, reflecting only a preferential activation weighting of each actuator. We refer to this weighting as the relative actuator 'strength' as varying the weighting of K_{a1} and K_{a2} induces the effect of the robot having stronger shoulder or elbow motion during the reaching task.

$$U_p = g^T \begin{bmatrix} K_{a1} & 0 \\ 0 & K_{a2} \end{bmatrix}^{-1} g \tag{4}$$

Steepest descent control, minimising for U_p (4), has been achieved by using $\nabla U_p = \frac{\partial U}{\partial q}$ for control: $\Gamma_p = -K_p \nabla U_p - K_d\dot{q}$. As in [4], stability is achieved through a damping term K_d. After inclusion of a posture space projection term $(I - J^T\bar{J}^T)$ the overall controller is therefore:

$$\Gamma = (J^T f) + (I - J^T\bar{J}^T)\Gamma_p \tag{5}$$

3.3 Sliding Mode Controller

Initial implementation of the task/posture controller produced poor performance in the robot with significant steady state error, particularly over small changes in the target height ($\nabla X_{y0} \leq 0.1m$). Inclusion of an integral term in the controller (6) improved this somewhat, though performance remained unacceptable for practical application (Figure 3).

$$f^\star = -K_x(X_y - X_{y0}) + \int_0^t (K_i X_y(\tau) - X_{y0})\, d\tau - K_v\dot{X}_y \tag{6}$$

The performance limitations over a limited demand range could be attributed to model parameter mismatch and un-modelled actuator dynamics that produced

friction and sticktion. Although the integrator improved task motion, the inability to overcome sticktion for small movements meant that it was deemed as insufficient. For small demand changes, friction and sticktion are difficult to overcome due to the low peak velocity / acceleration of the system. It is well known that sticktion / friction cannot be compensated by simple integral action and nonlinear control is required [8]. Thus the introduction of a sliding mode term provided a much better approach to reducing error in X_y.

$$f = \hat{\Lambda} f^{\star} + \hat{\mu} + \hat{p} + \hat{\Lambda} u_{sl} \tag{7}$$

The sliding mode controller is based on the sliding surface s:

$$\begin{gathered} u_{sl} = \tfrac{-K_{sl} s}{|s| + \delta}, \quad s = \dot{X}_y + K_s X_y, \\ K_x = -(K_s - K_v) K_s, \quad K_v = (K_x + K_s^2)/K_s, \quad K_i = 0, \end{gathered} \tag{8}$$

The effectiveness of the sliding controller depends on the speed at which u_{sl} is able to reach the sliding surface, for which $-K_s$ has to be a stable system matrix. The stable matrix $(K_s - K_v)$ influences the linear sliding mode reaching dynamics of the controller, while $K_{sl} > 0$ is chosen large enough to guarantee robustness [6].

3.4 Tracking Demand Signal

Though the sliding mode controller improved the robot's performance significantly, the speed at which the error was eliminated led to a jerky motion in X_y. Smooth motion is a prominent feature in human motion studies [7], therefore this jerk had to be overcome. Passing the step input demand through a second order filter (9) smoothed the demand signal in order to produce a smooth resulting motion.

$$X_{y0}(s) = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} X_{yd}, \tag{9}$$

Where X_{yd} is the original, unfiltered demand signal. This introduces additional dynamics (in that $\dot{X}_{y0}$ & $\ddot{X}_{y0} \neq 0$) that must be accounted for in the task controller:

$$f^{\star} = -K_x(X_y - X_{y0}) - K_v(\dot{X}_y - \dot{X}_{y0}), \quad f = \hat{\Lambda} f^{\star} + \hat{\mu} + \hat{p} + \hat{\Lambda} u_{sl} + \hat{\Lambda} \ddot{X}_{y0}, \tag{10}$$

4 Results

4.1 Human-Like Motion

The folding of the elbow to reduce effort has produced a vertical reaching motion that is comparable with that observed in volunteers (Figure 1). Particular features have been replicated, i.e. the initial folding of the elbow to reduce the arm's moment of inertia followed by straightening out at the end of the motion.

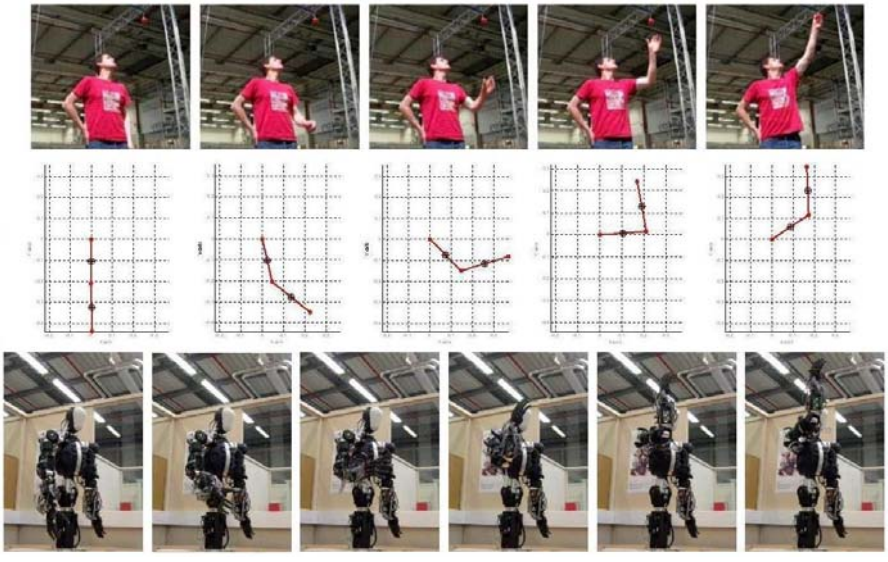


Fig. 1. Comparison of human reaching motion with simulated and physical robots

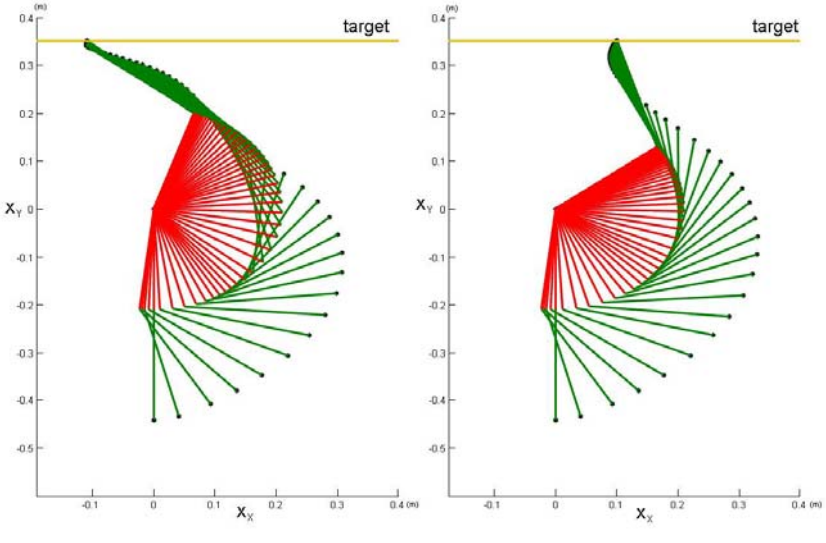


Fig. 2. 2 dof redundant reaching task simulations with a) Strong Shoulder Actuator b) Strong Elbow Actuator. The arm begins orientated in the direction of gravity.

The controller has been applied to two robots, a limited dof manipulator arm and a humanoid torso. In both cases the reaching motion is aesthetically pleasing and significantly different from the motion produced by a typical inverse kinematic formulation [2] (in which joints level control leads symmetric position and velocity profiles).

4.2 Simplified Posture Controller

The effect of the simplified posture controller can be visualised using a dynamic simulation in Figure 2. This shows a trace of the simulated 2dof manipulator reaching for a height of 0.35m using shoulder and elbow actuators. In Figure 2.a the parameters $K_{a1} = 100$ & $K_{a2} = 30$ lead to a stronger shoulder while in Figure 2.b $K_{a1} = 30$ & $K_{a2} = 100$ create a stronger elbow.

The motion pattern produced in Figure 2.b (i.e. the *strong elbow* motion) has the most similarity to the reaching pattern of a human. The elbow is flexed inwards early on in the motion leading to reduced inertia, there is also a straightening motion at the end of the movement where the forearm is extended. Measures of the effort variable U_p identified that the *strong elbow* motion achieved a peak effort measure $\sim 38\%$ less than the *strong shoulder* condition (where a greater portion of the lifting motion is produced by the shoulder).

4.3 Performance

Figure 3 provides a comparison of the various controllers in practical experiments with an anthropomorphic robot arm (using elbow and wrist joints instead of shoulder and elbow). The *'Original'* plot corresponds to the controller as detailed in [4] with task gains $K_x = 600$ and $K_v = 22$. *'Integral'* adds the integral term

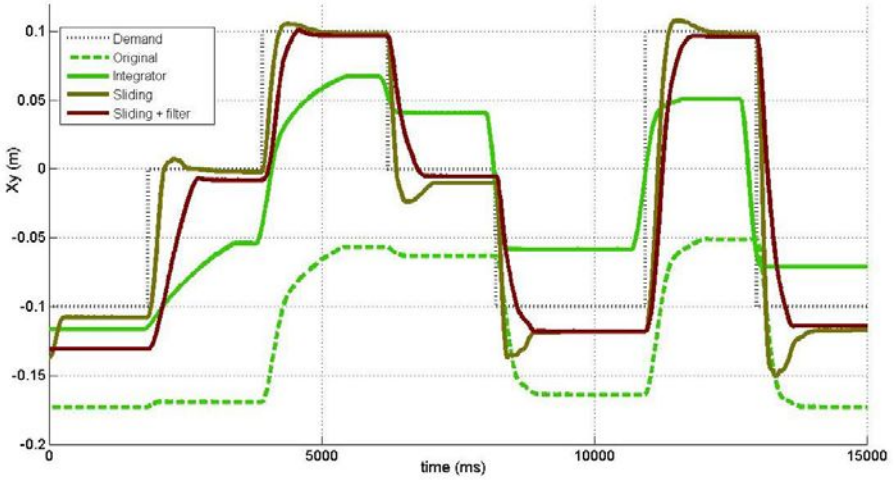


Fig. 3. Response of a physical robot system to X_y demand changes of 0.1m and 0.2m

detailed in (6) using $K_i = 49$. *'Sliding'* is the task sliding mode controller with unfiltered demand signals shown with gains $K_x = 85$, $K_v = 3.83$ and $K_s = 7.5$. This controller produced jerky, unnatural movement. The *'Sliding + filter'* plot shows the sliding mode task controller with the filtered demand signal (the filter has a time constant of $t_s = 0.55s$) with gains $K_x = 85$, $K_v = 12$ and $K_s = 5$ (note the reduction in K_x from the 'original' controller).

In all experiments the arm began motion from an initial configuration with the elbow bent. Repeating the movement process using the sliding + filter controller with the arm straight led to very similar task space performance despite very different joint trajectories. In all cases the posture gains $K_p = 0.1$ and $K_d = 0.5$, while the activation matrix gains were $K_{a1} = 10$ and $K_{a2} = 5$ (to produce the 'strong elbow' case). In plots involving the sliding mode controller: $K_{sl} = 220$.

4.4 Compliance

An interesting and practical feature of the implemented control scheme is that the redundant task and non-joint level control leads to a compliant system. As the task is achievable through a number of joint configurations a physical disturbance at one joint will cause a compensatory movement at the other joint in order to maintain the desired height. This is demonstrated in Figure 4 (using elbow and wrist joints) where the target fingertip height is recovered via a new arm configuration. To illustrate the accuracy of the controller a fixed red laser marks the target height. This target has been achieved in frames 1, 3 and 4.

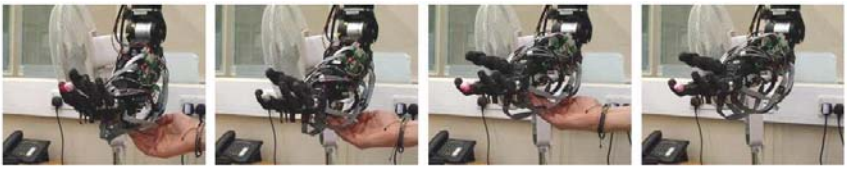


Fig. 4. Compliance: height of end effector (white fingertip) is maintained in presence of physical disturbance

5 Conclusion

A number of modifications have been made to the operational space framework proposed by Khatib/De Sapio et al in order to improve it's performance when applied to physical robots. The addition of a sliding mode controller has improved task space performance, while filtering input demand signals has maintained smooth and human like motion. The controller has been implemented on physical humanoid robots which have convincingly synthesised human-like reaching motion using a gravity based, effort minimising posture controller. Due to the redundant nature of the reaching task, the physical system has also demonstrated compliance in the presence of physical disturbances.

References

1. Bicchi, A., Peshkin, M.A., Colgate, J.E.: Safety for physical human-robot interaction. In: Springer Handbook of Robotics, pp. 1335–1348 (2008)
2. Craig, J.J.: Introduction to Robotics: Mechanics and Control, 3rd edn. Pearson Prentice Hall, Upper Saddle River (2005)
3. De Sapio, V., Warren, J., Khatib, O.: Predicting reaching postures using a kinematically constrained shoulder model. Advances in Robot Kinematics 3, 209–218 (2006)
4. De Sapio, V., Warren, J., Khatib, O., Delp, S.: Simulating the task-level control of human motion: a methodology and framework for implementation. The Visual Computer 21(5), 289–302 (2005)
5. Delp, S.L., Anderson, F.C., Arnold, A.S., Loan, P., Habib, A., John, C.T., Guendelman, E., Thelen, D.G.: Opensim: Open-source software to create and analyze dynamic simulations of movement. IEEE Transactions on Biomedical Engineering 54(11), 1940–1950 (2007)
6. Edwards, C., Spurgeon, S.: Sliding Mode Control: Theory and Applications. Taylor and Francis, London (1998)
7. Flash, T., Hogan, N.: The coordination of arm movements: An experimentally confirmed mathematical model. Journal of neuroscience 5, 1688–1703 (1985)
8. Hermann, G., Ge, S.S., Guo, G.: Discrete linear control enhanced by adaptive neural networks in application to a HDD-servo system. Control Engineering Practice 16(8), 930–945 (2008)
9. Khatib, O., Sentis, L., Park, J., Warren, J.: Whole body dynamic behaviour and control of human-like robots. International Journal of Humanoid Robotics 1(1), 29–43 (2004)
10. Lacquaniti, F., Soechting, J.F.: Coordination of arm and wrist motion during a reaching task. The Journal of Neuroscience 2(4), 399–408 (1982)
11. Uno, Y., Kwato, M., Suzuki, R.: Formation and control of optimal trajectory in human multijoint arm movement. Biological Cybernetics 26, 109–124 (1989)

Topology Control in Large-Scale High Dynamic Mobile Ad-Hoc Networks

Mohamed El-Habbal[1], Ulrich Rückert[1], and Ulf Witkowski[2]

[1] System and Circuit Technolog, Heinz Nixdorf Institute, University of Paderborn, Fürstenallee 11, 33100 Paderborn, Germany
[2] Electronics and Circuit Technology, South Westphalia University of Applied Sciences Soest, Germany
witkowski@fh-swf.de, {habbal,rueckert}@hni.upb.de

Abstract. We present our contribution in projects related to ad-hoc networking using different routing protocols and hardware platforms, showing our results and new solutions regarding topology control and routing protocols. We mainly focus on our work in the GUARDIANS EU-project, where as a main disaster scenario a large industrial warehouse on fire is assumed. The paper presents the simulation results for the routing protocols ACR, DSR and EDSR, as well as the implementation of down-scaled demos for supporting the autonomous team of robots as well as the human squad team with robust communication coverage. Various hardware platforms were used in the demos for distance measurement, based on laser range finder and radio communication with time of flight analysis.

Keywords: topology control, ad-hoc networks.

1 Introduction

Mobile ad-hoc networks (MANETs) are communication networks that do not rely on fixed, pre-installed communication devices like base stations or pre-defined communication cells. Manets are wireless networks consisting of mobile nodes which are characterized by their decentralized organization and the potentially high dynamics of the network structure. Therefore, Manets are ideally suitable for applications with multi-robot systems [1].

Wireless communication is essential in MANETs to support command and data exchange between mobile nodes. Depending on the application and key parameters like network size, data rate, coverage range, power consumption and application scenario (indoors / outdoors), a single or a combination of several communication standards should be chosen to suit all requirements.

One of the most promising applications of a multi-robot system is to assist humans in urban search and rescue (USAR) scenarios in the aftermath of natural or man-made disasters. As an example, our workgroup is contributing in the GUARDIANS project funded by the Sixth Framework Program of the European Union (no. 045269), where a main disaster scenario of a large industrial warehouse on fire is assumed [2]. In this scenario, black smoke may fill large space of the warehouse which makes it very difficult for the firefighters to locate themselves and orientate in the building.

J.-H. Kim et al. (Eds.): FIRA 2009, LNCS 5744, pp. 239–250, 2009.

Therefore, ad-hoc network communication system based on the mobile robots as communication nodes is deemed suitable because it can offer a robust communication infrastructure. Additionally, in order to increase the area coverable by the fire fighters, the ad-hoc network has to provide position data to support localization of the mobile robots and humans, which might be of great importance to guide the humans and robots to specific targets and locations or to quickly exit the search area.

Various topology control methods were tested on simulators and implemented on real hardware to support the required features in the GUARDIANS project. An example for that is the HCR (Hierarchal Cluster-based Routing) [3], where a cell based grid with master nodes in each cell forms the basic structure of the network. Some nodes formed by special robots act as beacons to uniformly span the network. These robots will act as reference points when positioning other mobile robots or humans and at the same time form the infrastructure to support communication all over the search area. A combination of distance and radio signal quality measurements as well as dedicated swarming behaviors of the robots ensure suitable distribution of the robots even in the presence of walls for example that are impervious to radio signals. After setup, the network should be able to offer different services, like message routing, localization and service discovery.

Different routing protocols were to be tested and evaluated for the project, some are proactive, others are reactive and hybrid. These were to be evaluated on both a network simulator as well as on real hardware platforms, and then comparing between their performances and hence choosing the protocol the best fits into the GUARDIAN scenario. Some of these protocols are still under evaluation while others are implemented and tested.

The paper is organized as follows: Section 2 gives an overview on the various routing protocols considered in the GUARDIAN project. In section 3 experiments and real down-scaled demos are described, and simulation results are presented in section 4. Finally, the paper is concluded in section 5.

2 Topology Control in the GUARDIANS Project

The main objective of the communication system in the GUARDIAN project is to provide the two main teams; the autonomous swarming team and human squad team, with robust communications links. Conditions like high dynamicity of robot and fire fighter movements, as well as reflective metal surfaces and hazardous materials in the scenario, could cause the communication links to become unstable and more likely to fall. Hence, several routing techniques are studied to best serve in such conditions and provide robust, fast adaptable communication coverage, especially to provide a reliable connection between the fire fighter and the base station, as well as with other robots. These routing techniques can be mainly divided into two categories; infrastructure-less- and infrastructure-based routing. By infrastructure-based routing we mean that some nodes (in this case scenario these nodes are robots) are dedicated to act as relay nodes to support the communication coverage. This can be done either by sending commands to some of the existing robots inside the scenario to switch their role, hold their positions and act as relay nodes, or by sending some additional nodes inside the scenario area specially dedicated for the communication purpose from the beginning, as will be explained in the second part of this section.

2.1 Infrastructure-Less Routing

Several infrastructure-less routing techniques are developed for mobile ad-hoc networks. These are mainly categorized as proactive, reactive, hybrid and geographic. Proactive routing maintains fresh lists of destinations and their routes by periodically distributing routing tables throughout the network. This has the benefit of fast connection establishment to any node when required, but has the drawbacks of high energy consumption, respective amount of data for maintenance, and slow reaction on restructuring and failures. One example of proactive routing that we used and tested in the GUARDIAN project is the OLSR (optimized link state routing) [4], where fewer nodes, called MPRs (Multi Point Relays) are selected for forwarding messages to the whole network. These are selected according to shortest path or best quality methods, and thus having the advantage of reduced message flooding; hence less interference and data losses, besides the reduced message overhead and size, as shown in figure 1.

Reactive routing is currently under evaluation and testing for usage in the GUARDIAN project. In this type of routing, a route is found on demand by flooding the network with Route Request packets. This has the drawbacks of high latency time in route finding, besides that the excessive flooding can lead to network clogging, but on the other hand, it has the benefit of saving energy from unnecessary flooding and reducing probability of interference and collisions.

One example for reactive routing that is currently under evaluation is the DSR (Dynamic Source routing) [5] and the EDSR (Enhanced-DSR). DSR is considered to be a basic and simple reactive routing protocol where the source originates the route request to the destination (without the need of acknowledging in the forward process). Once the destination is reached, it sends back an ACK to the source (backward path). If the source does not receive an ACK within a specific time window, it initiates another search. If the link is for some reason broken, a simple method is to re-initiate another search and request a new route to the destination, instead of trying to maintain and fix the existing link. Both the DSR and EDSR are described in more detail in the simulation results section.

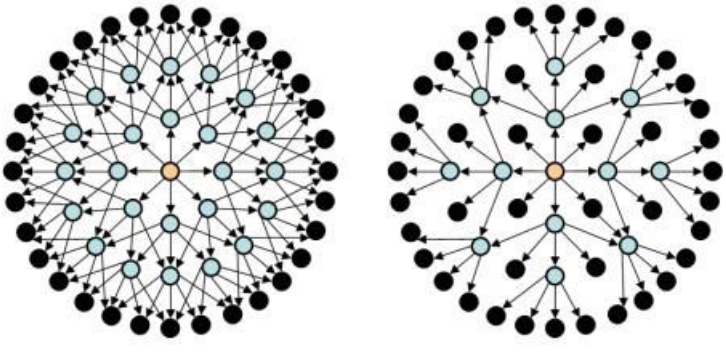


Fig. 1. Comparison between message flooding in normal and OLSR protocols

2.2 Infrastructure-Based Routing

In Infrastructure-based routing some nodes (called relay nodes) are dedicated to take over and be responsible for the communication process. In this kind of routing, some techniques partition the scenario area into clusters, and for each of these clusters a node (called cluster-head) is responsible for the incoming and outgoing messages inside, while remaining steady in their positions without critical movements as possible. This kind of routing is of interest in the GUARDIAN project since it provides very stable and robust coverage and is immune to high dynamicity in network topology where a robust link between the fire fighter and the base station has highest priority to remain connected. An interesting example would be the CGSR protocol (Cluster-head Gateway Switch routing) [6], where cluster-heads and gateways are used for routing messages (similar to OLSR).

Each node stores two tables; cluster member table, containing cluster-head ID closest to all other destinations, and DV-routing table, containing the next hop to each destination. This has the advantage of reduced message flooding and packet size, and is presented in figure 2.

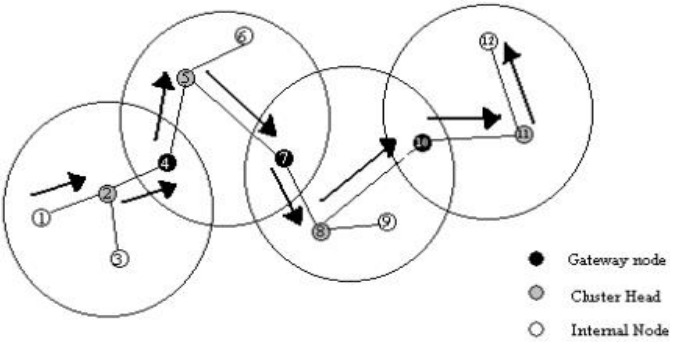


Fig. 2. Routing in CGSR

Another type of Infrastructure-based routing is the HCR (Hierarchal cluster-based routing), which is similar to the known HSR (Hierarchal state routing) [7], but with the difference that, instead of electing some nodes to act as cluster-heads from the nodes inside the ad-hoc network, special nodes are used and chosen from the beginning for this purpose, where they first enter the scenario area and distribute themselves inside in a uniform manner as possible, as shown in figures 3 and 4.

This gives the opportunity of equipping these nodes with the necessary communication platforms to support best coverage, while using a heterogeneous team of robots where other robots are not necessarily equipped with the same powerful communication platform. These infrastructure robots form a cell based network, in such a way that the communication network can robustly operate and that position data can be provided relative to a main reference node placed at the entrance of the scenario area. Since the actual exact situation inside the scenario is not available, these robots should try to distribute themselves to cover the search area and possibly

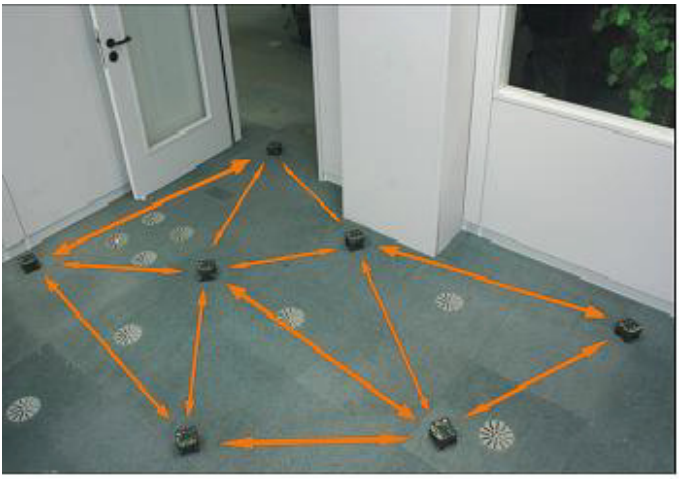


Fig. 3. Robots are spanning a communication network in a triangular mesh after they drove into a room (downscaled scenario)

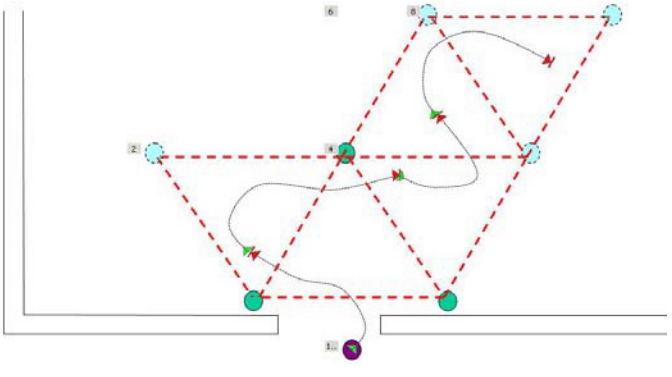


Fig. 4. Fire fighter following while forming dynamic triangulations to ensure robust communication coverage

position themselves in a uniform manner. Our proposed way to do that in a simple and effective way is to distribute them first in a mesh of triangles with three different modes to avoid obstacles in their way, starting from equilateral triangles mode (60 degrees triangles), 120 degrees triangles mode in case of obstacles and ending up forming a line (180 degrees triangles) in the worst case, in cases like long passages and narrow corridors for example. After this initial distribution the network's topology can be adapted according to the current situation. Simulation and evaluation of the HCR protocol's performance is currently under development for the GUARDIAN project, while down-scaled demos were done to test the infrastructure distribution process, as well as the fire fighter following as will be described in detail in the demos section.

3 Experiments and Results

3.1 Laser-Based Triangular Distribution of Infrastructure Nodes

This demo is to show how a group of mobile robots (beacons) detect the rough location of the fire fighter and distribute themselves with other communication nodes in triangulation forms following the fire fighter's path. This is to guarantee robust communication coverage between the fire fighter and base station, as well as helping the other swarming team for map building and area exploration. This is shown in figure 4.

The position of the beacons assuming line of sight communication along the triangles and local scanning of the environment enables rough maps of the environment. The positioning of the nodes (beacons) is based on distance measurements and odometry. This part includes circumnavigation of obstacles by robots.

The demo goes as follows: Three robots are present at the beginning close to the entrance, as marked in green in figure 4. These robots represent the core of the triangulation distribution of communication nodes, which are equipped with laser sensors to detect the rough location of firefighter. The laser sensor used in this demo is the IFM single beam diffusion laser sensor. Hence, the rough location of the fire fighter will be detected according to the laser rotated angle. As shown in the fire fighter's path drawn in figure 4, the fire fighter first enters the scenario area, which means he is inside the core triangle formed by the 3 green nodes in figure. If he moves left to the left side triangle (left sector), another (fourth) beacon will move to the dashed light blue position on the left as shown in figure. As the fire fighter moves back to the right sector, another beacon will go to its desired position to complete the formation of the triangulation distribution. The fire fighter is equipped with Laser detectors to detect the Laser beam and send messages back to the scanning beacon to estimate his position. Probably each sector (triangle) will be scanned by only one robot at a time, in order to know the ID of the scanning robot to which messages (acknowledgements) are sent back from the fire fighter. The range (angle limits) of each scanning robot is set according to its position in the scenario.

3.2 Infrastructure-Based Routing with Laser-and Radio-Based Fire Fighter Following

This demo is to demonstrate how a link between the fire fighter and the base station outside the scenario warehouse is established. The relay nodes following the firefighter will be equipped with Hokuyo Laser sensors to follow other robots and the fire fighter. Minimum and maximum thresholds are taken into consideration for the movement control of the first team ("fire fighter follower team"); the first threshold is to avoid collision with the followed target, while the latter is to not to lose the followed target and to increase speed. Another threshold is used by the "relay team" which gives an order to stop the followed robot.

The demo is depicted in figure 5 and goes as follows: the "fire fighter follower team", shown in light blue in figure 5, will follow the fire fighter (shown in purple) in a line, where they are placed close to each other within the two thresholds explained

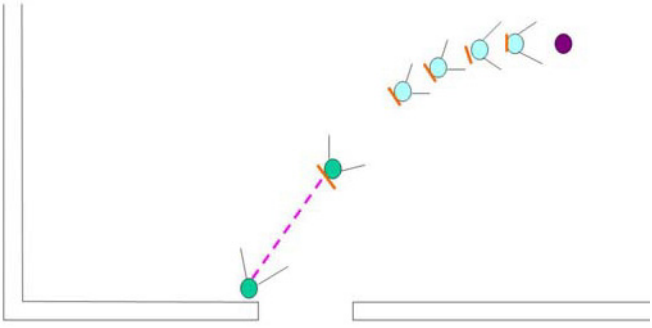


Fig. 5. Fire fighter following using Laser sensors

above. Each robot in this team will follow the preceding robot in front of it, except for the first robot which follows the fire fighter directly. IR-sensors, shown as orange plate lines in figure 5, are attached to the back of each robot for robot recognition as well as the fire fighter legs (or robot representing the fire fighter in this demo) for identifying the fire fighter. The "relay team", shown in green in the figure, will start with only one robot, which is the one acting as the main reference point at the entrance. It will start measuring the distance between it and the last robot in the first team using the Laser sensor. If this distance exceeds a certain threshold, this robot will order the latter to stop moving and act as a new relay node. Bluetooth and WLAN will be used as the communication standards between all robots and the base station (not shown in fig. 5) in this demo.

Another similar demo is also done but using a combination of ultrasonic sensors and Nanotron radio-based distance measurement sensors [8] instead of laser sensors that were used in the previous demo, as depicted in figure 6.

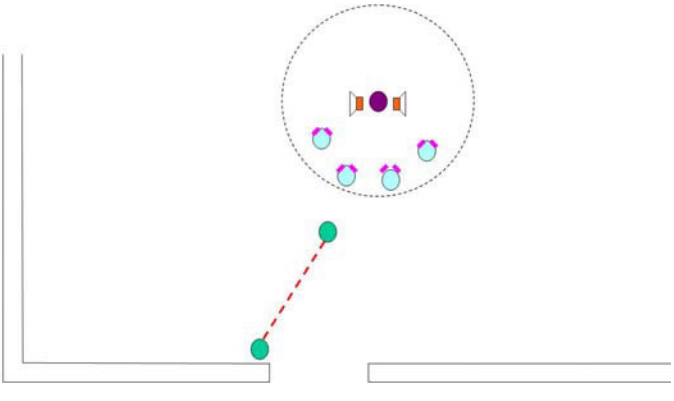


Fig. 6. Fire fighter following using Radio-based distance sensors and ultra-sonic-based swarm formation

4 Simulation Results

The simulator used for network protocols implementations and evaluation is the Netlogo multi-agent simulator, which uses its own object-oriented programming language similar to C^{++}.

A new technique, which is the ACR (Ant Colony Routing) [9], was first implemented on the Network simulator. The ACR algorithm used for communication protocols is mainly based on the idea of choosing not only one (normally best) route between a source and destination, but instead building up multiple independent routes. This is the same concept which ants use when searching for food where each ant leaves a trace from nest to food source which other ants can follow. Hence, it is not necessary that all other ants follow the same path, but some ants try other paths, and if succeeded, leave traces in turn. The more ants using the same path, the stronger this trace gets, which indicates that this route is reliable. The same idea goes for the ACR protocol, where multiple routes exist, and the weights of each path indicate the path quality, which is analog to the trace used by ants. This weight normally represents the number of hops, which the choice of best route is based on. The smaller the number of hops between the source and destination in a path, the more it is considered to be a better choice, as shown in figure 7. The advantage of ACR is that it gives an indication of congestion level of nodes used in paths between source and destination. If only one route exists, this means that this connection is vulnerable and there is a possibility that if a node fails, the connection could be interrupted. ACR provides more immunity against bad conditions that could affect the communication coverage, but does not really support high dynamicity in nodes movements, since searching for optimal and other possible routes costs much time and energy, and fixing interrupted links is complicated.

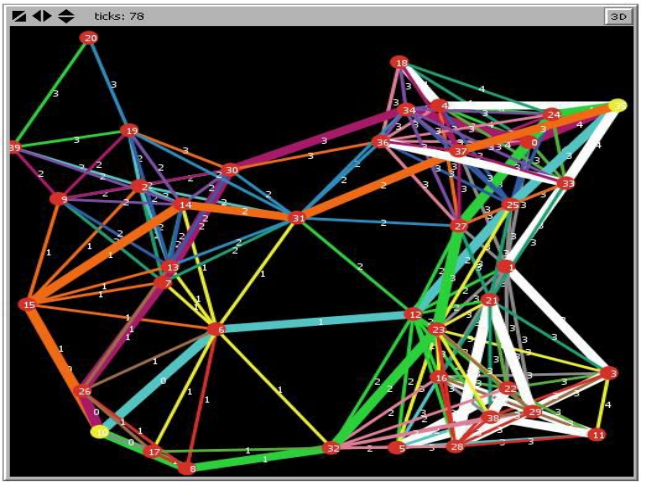


Fig. 7. Multiple independent routes in ACR

As stated in section 2, DSR protocol was also tested on the simulator. DSR has the advantage of simplicity and cost efficiency of route establishment, which was worth testing for providing robust coverage in a network where nodes are moving rapidly, as shown in figure 8.

Here is a summary of the simulation parameters used; 80 nodes distributed in a 40m x 40 m square area, moving with speed varying from 0 to 5 m/s, 115200 baud-rate, 13 m maximum coverage radius, packet sizes varying between 40 and 100 bytes, including a 20 byte overhead in each. For the tests, 4000 data bytes were to be sent from source and acknowledged by destination. Figure 9 shows the effect of increasing moving speed of nodes on the connection break down and hence packet loss. Once a

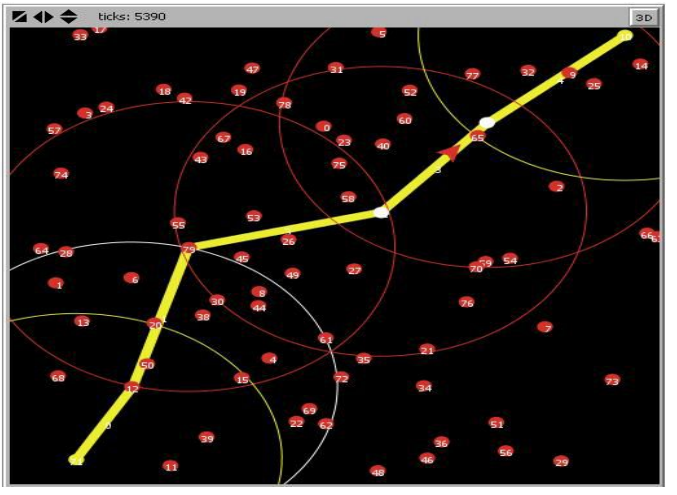


Fig. 8. Route establishment in DSR

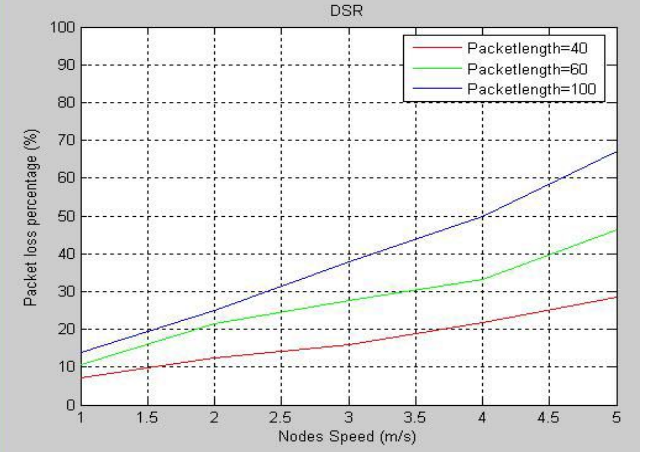


Fig. 9. Performance of DSR with various packet lengths and speeds

link is broken, a new Route-Request is performed, in order not to waste time trying to fix an existing broken link which might take more time compared to the fast search mechanism of DSR.

From the graph in figure 9 one can notice that as the packet size increase, the packet loss also increase, as it takes more time for a full round trip from source to destination and acknowledged back to source, in which the nodes would have moved further from each other and hence increasing the probability of connection failure. The source and destination are steady and do not move during all simulations in order not to change the distance between them and thus affect the simulation results.

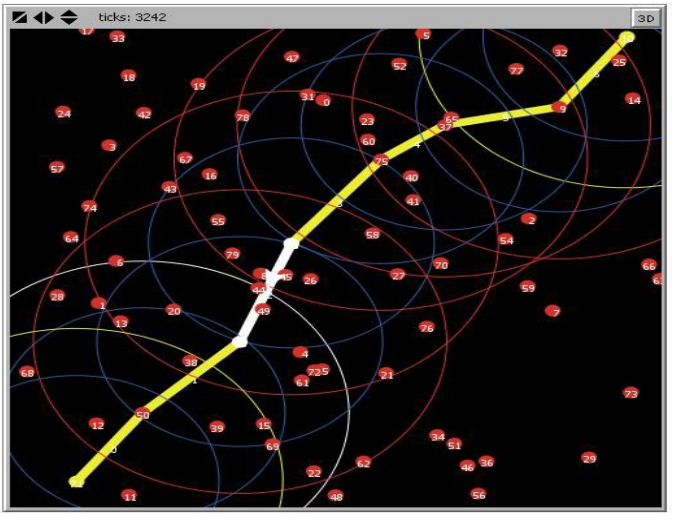


Fig. 10. Route establishment in EDSR

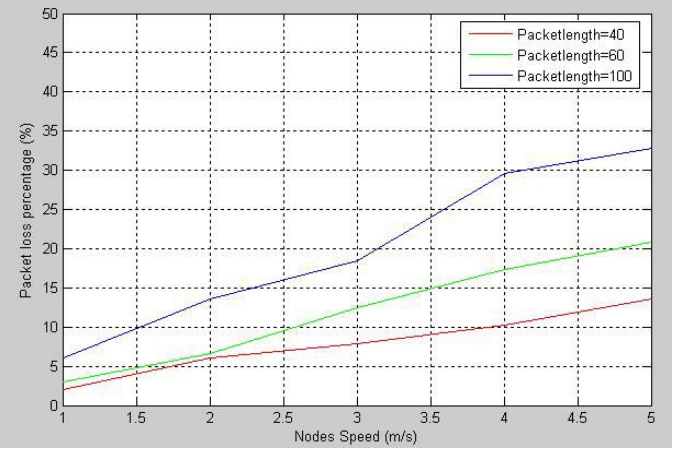


Fig. 11. Performance of EDSR with various packet lengths and speeds

Another test is done for the EDSR. EDSR (Enhanced-DSR) is similar to the traditional DSR but with a minor enhancement to achieve a more stable performance especially in high dynamic topologies. Since DSR chooses its best route based on minimum number of hops, it might happen that two neighboring nodes in the path lie very close to the edge of each others coverage, and hence the link between them becomes vulnerable to any movement of the nodes. EDSR on the other hand chooses a neighboring node that lies in an inner-radius smaller than the maximum coverage radius to give some room for a node to move before the link breaks down, and hence provide more stability and robustness in connections. This is shown in figure 10, where inner-radii are presented in blue. The performance of EDSR is depicted in figure 11. Tests were done with maximum coverage radius of 13m as in DSR, and inner-radius once with 11m and in another test with 9m. As the change in inner-radius affects the results very slightly, the results of only one of them are presented. Comparing the results of DSR and EDSR, one can notice that EDSR provides better performance with less packet loss for various packet lengths and node speeds.

5 Conclusion

We have focused on our work in the GUARDIANS project as a contribution in projects related to ad-hoc networking using various routing protocols that are tested in real down-scaled demos and evaluated on simulators. One of the key issues in this project is maintaining a robust communication system providing both communication between all team members and position data. Some methods in both infrastructure-less and infrastructure-based routing were presented, with the first having the advantage of simplicity and avoiding additional hardware, while the latter providing more stability, supporting robustness in high dynamic networks, and might offer additional useful features like positioning and localization as supported by HCR. DSR and EDSR were evaluated using simulation under real parameter assumptions, while others are still under evaluation to compare and choose the routing protocol that best fits in the GUARDIAN project.

Acknowledgment

This work was supported by the Sixth Framework Program of the European Union as part of the GUARDIANS project (no. 045269, www.guardians-project.eu).

References

1. Hermann, R., Husemann, D., Moser, M., Nidd, M., Rohner, C., Schade, A.: DEAPspace: Transient Ad-Hoc Networking of Pervasive Devices. In: Proceedings of the 1st ACM International Symposium on Mobile Ad Hoc Networking & Computing. IEEE Press, Boston (2000)
2. Witkowski, U., El-Habbal, M., Herbrechtsmeier, S., Penders, J., Alboul, L., Motard, E., Gancet, J.: Mobile Ad-hoc communication in highly dynamic environment optimized with respect to robustness, size and power efficiency. In: Proceedings of the International Workshop on Robotics for risky interventions and Environmental Surveillance (RISE 2009), Brussels, Belgium (2009)

3. El-Habbal, M., Witkowski, U., Rückert, U.: Mobile Ad-hoc Communication applied and optimized for disaster scenarios. In: Proceedings of Wireless Technologies Kongress 2008, Bochum, September 2008, pp. 25–34 (2008); SCT, ad-hoc network, GUARDIANS
4. Jacquet, P., Muhlethaler, P., Qayyum, A., Laouitl, A., Viennot, L., Clausen, T.: Optimized Link State Routing Protocol (OLSR), http://wiki.uni.lu/secan-lab/Optimized+Link+State+Routing+Protocol.html
5. Johnson, D., Maltz, D.: Dynamic source routing in ad hoc wireless networks. In: Imielinski, T., Korth, H. (eds.) Mobile Computing, vol. 353. Kluwer Academic Publishers, Dordrecht (1996)
6. Chiang, C.: Routing in Clustered Multihop, Mobile Wireless Networks with Fading Channel. In: IEEE Singapore International Conference on Networks, SICON (1997); Clusterhead Gateway Switch Routing (CGSR), http://wiki.uni.lu/secan-lab/Cluster-Head+Gateway+Switch+Routing+Protocol.html
7. Miller, E.H.: A note on reflector arrays (Periodical style—Accepted for publication). IEEE Trans. Antennas Propagat. (to be published); Hierarchical State Routing - Scalable Routing Strategies for Ad Hoc Wireless Networks, http://wiki.uni.lu/secan-lab/ Hieracical+State+Routing.html
8. Nanotron Technologies, Berlin (2007), http://www.nanotron.com
9. Liao, W.-H., Kao, Y., Fan, C.-M.: An Ant Colony Algorithm for Data Aggregation in Wireless Sensor Networks. In: International Conference on Sensor Technologies and Applications, pp. 101–106 (2007)

A Gait Generation for an Unlocked Joint Failure of the Quadruped Robot with Balance Weight

C.H. Cho, B.C. Min, and D.H. Kim

Department of Electronic Engineering, Kyung Hee University,
Seocheon-dong, Giheung-Gu, Yongin-Si, 449-701, Korea
{movielove61,minbc,donghani}@khu.ac.kr

Abstract. Assurance of a stability margin for a stabilized gait is the most important issue for the quadruped robot. Although various studies for dynamic stability of the quadruped robot have been studied, problems in which one of the legs has an unlocked joint failure haven't been relatively studied so far. In this paper, assurance of stability margin for the unlocked joint failure of the quadruped robot is suggested by using gait stabilization and a control method of the moment of inertia. Then, efficiency of BW (balance weight) will be experimentally verified by comparing the two types of robot; one is equipped with the BW, the other is not equipped with BW.

Keywords: quadruped robot, balance weight, joint failure.

1 Introduction

Although legged robots have more complex kinematic structure and less energy efficiency than wheel-based robot, legged robots have been developed a lot because of the high locomotion and environmental adaptability on uneven terrain among the multi-legged robots field. The studies on the quadruped robot are more dynamically stable and efficient than other robot.

Most studies in the quadruped robot focused on the stability or the gait generation. For example, Kumar[1] suggested force distribution method for walking. Hiroshi[2] experimented the quadruped robot on unknown irregular terrain with automatic posture control. Also, Fukuoka[3] presented adaptive dynamic walking on irregular terrain using a neural system model. Berkemeier[4] and Takao[5] suggested running method of a quadruped robot.

On the other hand, quadruped robot which has unexpected problems with one of the four leg has less been studied so far. In case of the locked joint failure, Yang[6] proposed stabilized gait generation method for the locked joint failure that one of the four legs still had the ability to support the body, but didn't have any ability to move the body. For the unlocked joint failure, Shigoe[7] suggested kick-and-swing movement, active shock absorbing method and minimum energy gait planning.

The purpose of this paper is to propose a method to control the quadruped robot by using a balance weight (BW) even if one of the four legs loses all the abilities such as supporting or moving the body.

J.-H. Kim et al. (Eds.): FIRA 2009, LNCS 5744, pp. 251–261, 2009.

Section 2 provides the modeling of the quadruped robot with BW (balance weight) on an unlocked joint failure. Adjustment of center of mass using BW (balance weight) is introduced in Section 3. Stabilization methods of the quadruped robot are suggested in Section 4, and comparisons between the model with BW and without BW on the same condition are presented and the efficiency of the BW is proved in Section 5. Some concluding remarks are given in Section 6.

2 Modeling of the Quadruped Robot with Balance Weight on Unlocked Joint Failure

Fig. 1 shows the quadruped robot with BW (balance weight) for the unlocked joint failure. Leg 0 (dotted line) is assumed to have an unlocked joint failure and BW is placed at the center of the robot body. BW has two links. Joint 1 is rotated to the horizontal direction and joint 2 is rotated to the vertical direction.

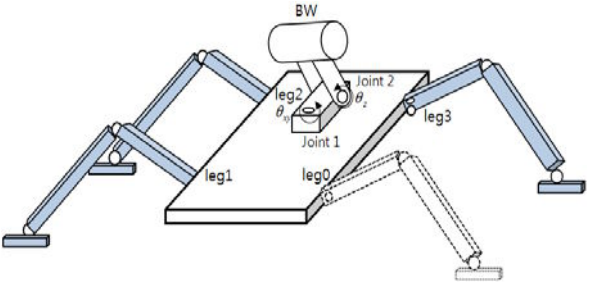


Fig. 1. Modeling of the quadruped robot with a BW on the unlocked joint failure

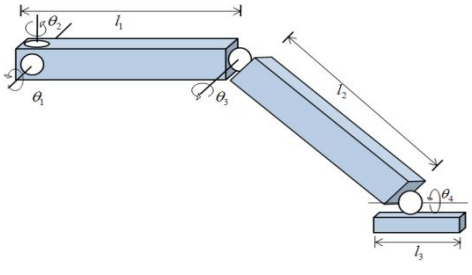


Fig. 2. Model of a leg

Fig. 2 is a perspective view of a leg attached to the considered quadruped robot. Each leg is composed of four joints. The upper link is connected to the body via two active revolute joints: joint1 (θ_1) which is utilized to lift a leg is vertical with the ground and joint 2 (θ_2) which is utilized to move a leg forward is parallel with the ground. Joint 4 (θ_4) is designed to make the coefficient of friction stronger with the ground while the robot is tilting on the movement.

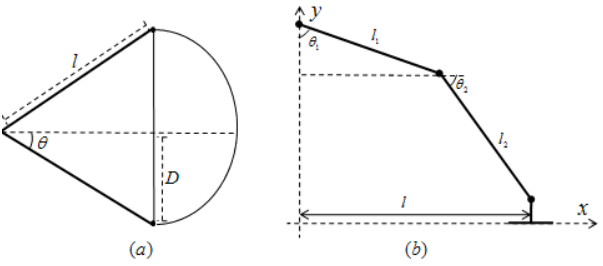


Fig. 3. Reachable area of the each leg (a) Working area (b) Lateral view of a leg

In Fig 3, Reachable area of the each leg is shown. Whole length (l) of a robot leg is made of three links. The last link of a leg can be omitted because it doesn't affect the total length (l). Total length of a leg is defined as follow.

$$l = l_1 \sin\theta_1 + l_2 \cos\theta_2 \tag{1}$$

Total reachable area of a leg (2D) is shown in Fig. 3 (a) and defined as

$$\begin{aligned} 2D &= 2l\sin\theta \\ &= 2(l_1 \sin\theta_1 + l_2 \cos\theta_2)\sin\theta \end{aligned} \tag{2}$$

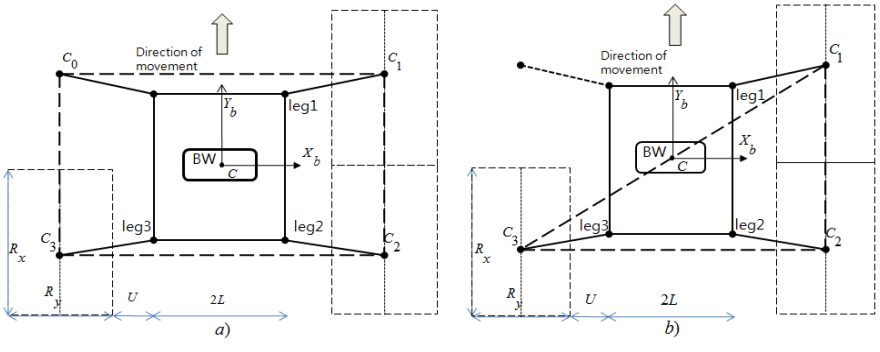


Fig. 4. Two-dimensional model of a quadruped robot (a) Stable center of mass and reachable area of each leg (b) Unstable center of mass and reachable area of each leg

Fig. 4 shows two-dimensional models of the quadruped robot and projected reachable area of each leg. It is assumed that X_b-Y_b is X-Y axis based on the robot body, Y_b is supposed as same direction as the movement of the quadruped robot. Dashed rectangles are working areas. C_i (i=0, 1, 2, 3) is the initial positions of each leg which is able to move anywhere in the reachable area of R_x, R_y.

Fig. 4 (a) represents center of mass(C) in normal situation. In this case, dashed rectangle line contacts with foothold of each leg (C_i (i=0, 1, 2, 3)) and the ground. C, the crossing between leg 0, 2 and leg 1, 3, is the center of mass and the origin of the robot body's coordinate. Although one of each leg is lifted to generate the gait for

moving the body forward, other three legs can stably support the body, thus a stable gait generation is possible even if the center of mass is changed in the dashed rectangle line.

On the other hand, Fig. 4 (b) represents center of mass(C) when one of the four legs occurs an unlocked joint failure. To make the robot walk stably, center of mass must be placed on the dashed triangular line, formed by the position of the three legs. If the center of mass is closed to one of the dashed triangular line, the robot can easily lose the balance by either the moment of inertia or external force, especially, when one of three legs is lifted to generate the gait. To make the center of mass inside of the dashed triangular line, the movement of BW is necessary while the robot is walking.

3 Center of Mass by Using Balance Weight (BW)

An unlocked joint failure is an unexpected event in which a leg of the four legs lost all its abilities such as supporting the body or moving the body. Immobilization of a leg enormously affects the stabilization and mobility of the quadruped robot.

If the quadruped robot with an unlocked joint failure can stay in the stability margin, the robot is able to move forward although one of the three legs lost all of its abilities.

There are various ways to keep the robot stable by securing the stability margin such as adjustment of the foothold position [8] and force distribution. In this paper, to secure the stability margin for the quadruped robot with an unlocked joint failure, gait generation method will be suggested by using BW.

For the simplicity of the modeling, several conditions of the quadruped robot are assumed as follow.

(1) The quadruped robot has the straight-line motion following Y_B axis.
(2) Foothold trajectories of the each leg are on the middle of reachable area.
(3) The slip of support legs doesn't occur.
(4) The center of body's mass is at the center point (C) of the body.

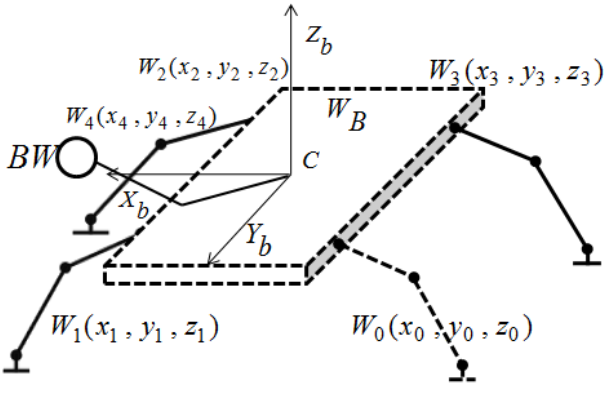


Fig. 5. Schematic of the quadruped robot with BW

In Fig 5, center of mass (C) of four legs and BW (W_i, i=0, 1, 2, 3, 4) on the free coordinate system on the three dimensional space is obtained as follow.

$$
\begin{aligned}
W &= W_B + \sum_{i=0}^{4} W_i \\
x_C &= \frac{1}{W}\sum_{i=0}^{4} W_i x_i \qquad y_C = \frac{1}{W}\sum_{i=0}^{4} W_i y_i \\
z_C &= \frac{1}{W}\sum_{i=0}^{4} W_i z_i
\end{aligned}
\tag{3}
$$

Where W_B is the weight of the robot body, W_i is the weight of the each leg and BW, W is the total weight of the robot. Center of mass of the robot can be easily calculated by using a simple equation (3). When the quadruped robot has an unlocked joint failure, it is difficult to move the position of the three legs which support the robot body. However BW can make the whole center of mass better by adjusting the position of it.

4 Stabilization Methods of the Quadruped Robot on Gait Stages

The quadruped robot that one of four legs has an unlocked joint failure in static gait encounters a serious problem to balance weight by the condition of terrain or subtle changes of robot joints [9]. This problem becomes more serious when one of three legs moves. If three of four legs support whole weight, in order word, center of mass is in triangular stability margin, two of four legs should support whole weight to walk stably when a leg is lifted.

In the following subsection, falling time of the robot, maximization and minimization method of the moment of inertia, and a gait generation method are suggested.

4.1 Falling Time of the Body When a Leg is Lifted

When quadruped robot doesn't have any leg problems, the robot can stably move forward. Even if a leg is lifted to take a next step, three supporting legs, forming triangular stability margin, is enough to support whole mass of the robot. However in case that the quadruped robot has an unlocked joint failure, the robot can't stably move forward because the robot needs at least three supporting leg to make a gait stable. Thus, we need to consider the falling time of the robot when the robot is supported only by two legs and the sufficient time for a gait, sequence of lifting and landing a leg.

In Fig 6, the height between the robot and the ground is set by the angle of each joint. The height of the robot is obtained as follow.

$$
\begin{aligned}
H &= h_1 + h_2 \\
&= l_1 \cos\theta_1 + l_2 \sin\theta_2
\end{aligned}
\tag{4}
$$

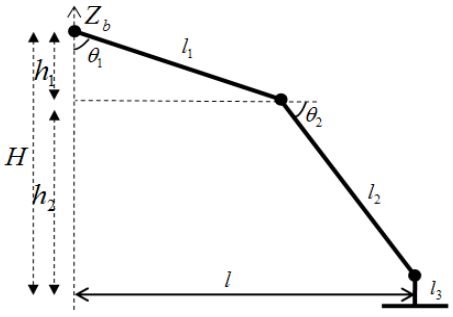


Fig. 6. Height of a leg

The minimum boundary height is supposed $1/3$ H of the total height because of the hardware limitation. The time that a leg is reached is derived as below by using free fall equation.

$$\frac{1}{3}H = H + v_0 t - \frac{1}{2}\alpha t^2$$
$$t = \sqrt{\frac{4H}{3g}} \geq T \tag{5}$$

Where v_0 (v_0 is set to 0) is an initial falling velocity, α is acceleration that substituted by acceleration of gravity (g). T is the time that a leg moves 90degree from the initial position. Therefore, it is necessary for a leg to move the next position of a leg before time T.

4.2 Maximization and Minimization Method of Moment of Inertia on a Gait Generation by Using BW

Moment of inertia, occurred by the movement of each leg's gait sequence and BW, is disturbance factor for the robot to stay in stability margin. However it is also necessary factor for the quadruped robot with unlocked joint failure to move.

When three legs support whole mass of the robot, the mass of the robot is separated differently. In Fig.5, leg1 supports half of whole robot weight because of the lack of leg0. On the other hand, each leg2, 3 support half of whole robot weight equivalently. That means, leg 1 needs much more movement of BW comparing with leg2, 3 in order to take a gait sequence. For the gait sequence of leg1, the BW is applied to maximize the moment of inertia by moving anti-direction of the leg's moving direction. In contrast, for the sequence of leg2, the BW is applied to minimize the moment of inertia by moving anti-direction of the leg's moving direction. The problem is the leg 3. The time for the gait sequence (lifting and landing a leg) of the leg 3 is not enough to complete. The leg3 supports half of whole robot weight with leg2 equivalently when whole legs support the robot.

However, if the leg 3 is lift to move, the robot is tilted to the direction of the leg3. Therefore, the BW should move to the right side of the robot in order to change the center of mass and it is applied to maximize the moment of inertia by moving same direction of the leg's moving direction in order to separate the robot weight.

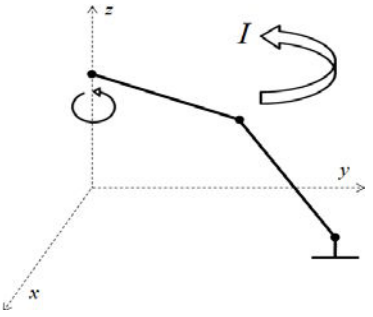


Fig. 7. Moment of inertia (*I*) of a leg

The mass per unit length of the robot is assumed identical. Supposed that a leg is moved anti-clockwise direction based on z axis in Fig. 7, the mass per unit length is

$$\lambda = \frac{W_{leg}}{l} \tag{6}$$

Where l is the length of a leg, W_{leg} is the total weight of the leg. If the length is dl

$$dw = \lambda dl = \frac{W_{leg}}{l} dl \tag{7}$$

By using Equation (6) and (7), the moment of inertia for the leg is derived as follow.

$$I = \int_0^l l^2 \frac{W_{leg}}{l} dl = \frac{1}{3} W_{leg} l^2 \tag{8}$$

4.3 A Gait Generation by Using BW

The gate generation for an unlocked joint failure of quadruped robot is suggested in this subsection. For the simplicity of the gait generation, following assumptions are needed.

(a) C_0 (left-front) leg is omitted by outer impact, so the leg (C_0) is lost all of its abilities to move the robot and support the ground.
(b) Only one leg is occurred unlocked joint failure in locomotion sequence.
(c) Moving speed of BW and legs is very short.
(d) The slip doesn't occur in the gait generation.
(e) The terrain is even.

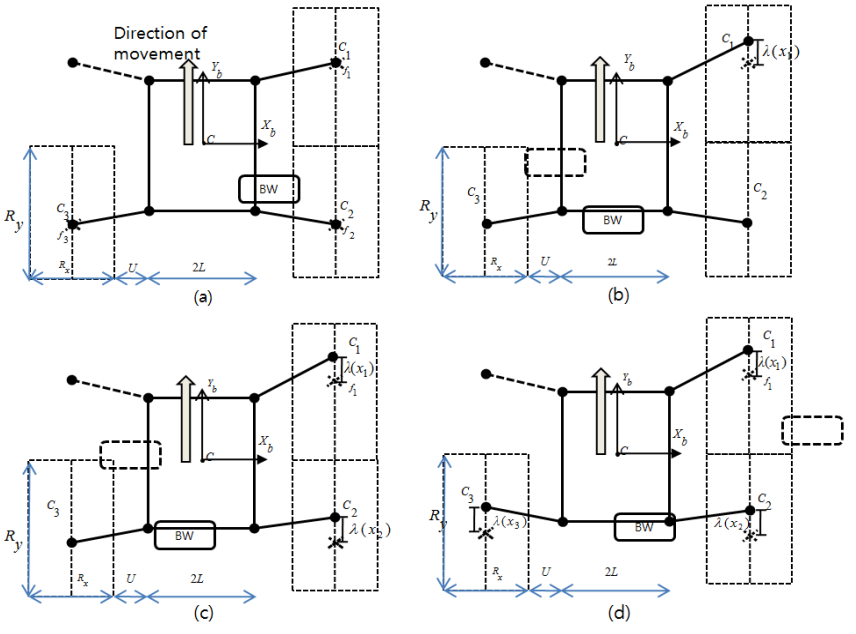


Fig. 8. Gait generation of the quadruped robot with BW

The gait generation is following steps below.

1) Initial foothold position:

In Fig. 8 (a) shows the initial foothold position. The center of robot's mass is adjusted by using BW to balance the whole weight of the robot in case that one of four legs is lost.

2) Sequence of foothold movement.

When the robot lifts one of the three legs to move the body, other two legs must maintain whole weight of the robot. At this time, center of robot's mass is placed on a straight line formed by two of the legs (leg1, leg3) if the leg 2 is moved. Then the moment of inertia is a disturbing factor for the robot to balance the whole weight when the leg (leg2) is lift to move forward. In this situation, BW is not only used to balance the center of the robot's mass, but also used to offset the moment of inertia by moving anti direction of the leg.

3) In Fig. 8 (b) – (c) – (d) show the position of BW and legs.

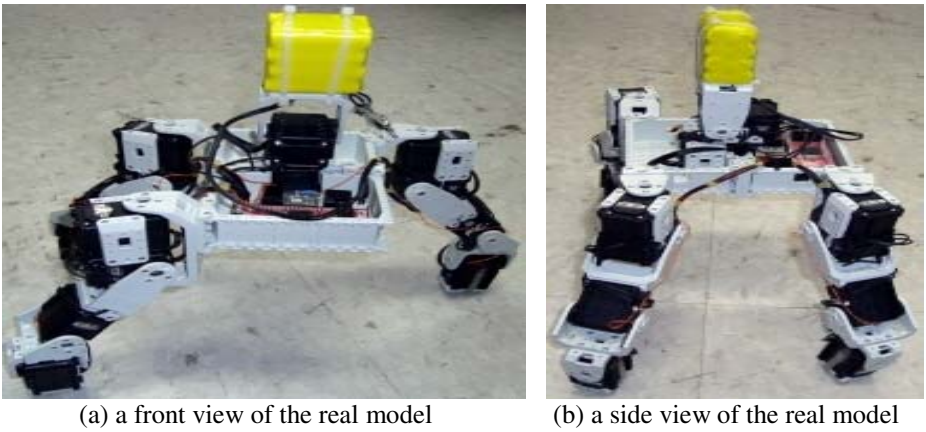
For a gait sequence of the leg 1 and leg 2, it is necessary to move the BW at the backside of the robot in order to maximize falling time of the robot. Moreover, BW is applied to minimize the moment of inertia by moving anti-direction of each leg's moving direction in order for the robot walk stably. On the other hand, in Fig. 8 (d), the stability margin is set on the straight line, formed by leg 1 and leg 2, when leg 3 is lifted. In this case, the BW should move to the right side of the settled stability margin in order to have sufficient falling time for the gait

sequence of leg 3. Furthermore, the moment of inertia should be faced to the right side of the robot. Therefore, the BW is applied for the leg 3 to maximize the moment of inertia. To have much time for a gait of leg 3, it is required for the BW that whole of the robot weight is inclined to the point between leg 1 and leg 2.

4) All legs go back to initial position (a) to move the body after finishing the sequence of foothold movement.

5 Experimental Results

Fig. 9 shows the real hardware model for the experiment. For the experiments, the robot equipped FSR(Force Sensing Resistor) sensors on each leg and two tilt sensor on the body. The battery, occupying high percentage of whole robot weight, is utilized for the BW. Left front leg is omitted to simplify experiments. Specific parameters are shown in Table. 1.

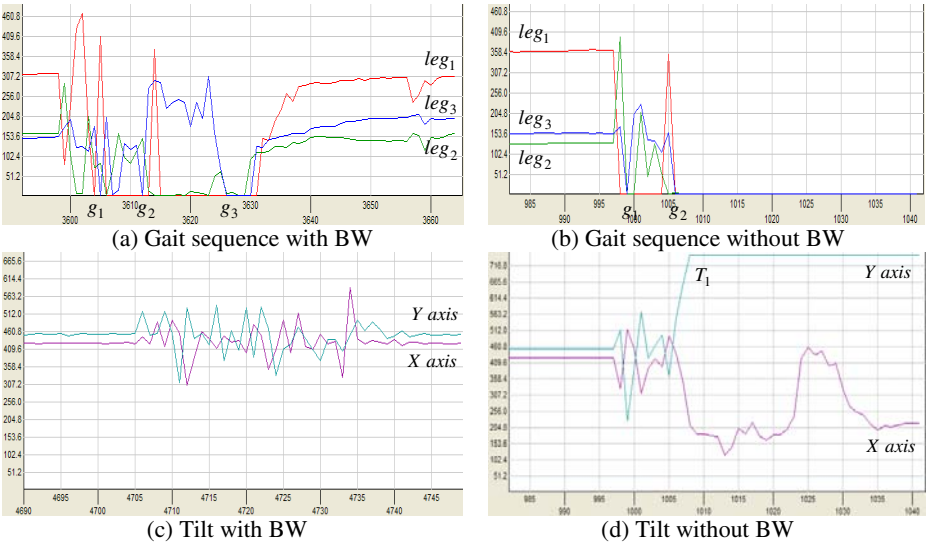


(a) a front view of the real model (b) a side view of the real model

Fig. 9. A real model for experiments

Table 1. The parameters of the quadruped robot

	Parameter	
Total weight	1640 g	
Height	35.0 cm	Including length of BW (15.0 cm)
Diameter	34.0 x 22.0 cm	
Length of link L_1	9.5 cm	
Length of link L_2	7.0 cm	
Weight of the BW	480 g	
Motor Speed	0.196 sec/ 60 deg	



(a) Gait sequence with BW (b) Gait sequence without BW

(c) Tilt with BW (d) Tilt without BW

Fig. 10. Experimental results

The experimental results are shown in Fig. 10 (a) – (d). X-axis indicates the data that simulation program is received. Y-axis indicates the intensity, measured by FSR sensors.

Fig. 10 (a), (b) show the pressure of each leg. In Fig.10 (a), the whole weight of the robot is distributed. Although the pressure, applied to the leg 2 and leg 3, is quite similar, the pressure, applied to the leg 1 is higher than that of the leg2, leg3 because of the lack of a leg (leg 0). In addition, the quadruped robot with BW completes the gait sequences and the sequence of a gait (lifting and landing a leg) is shown at points (g_1, g_2, g_3 in Fig10 (a)). On the other hand, in the Fig.10 (b), the robot without BW couldn't continue the gait sequence because the robot is fallen over at the point g_2. Fig.10 (c)-(d) is shown the tilt of the robot. Even though the quadruped robot swung on the gait sequence, the result shows that the robot completed the gait sequence without falling over by using BW. However, in the Fig. 10 (d), the robot is fallen over at the point T_1, a gait sequence of leg 2.

6 Concluding Remarks

When a quadruped robot has an unexpected event such as joint failure, the locomotion ability of the robot is seriously damaged. To solve this problem, in this paper, a gait generation with balance weight (BW) is suggested. As shown in experimental results, the quadruped robot for an unlocked joint failure is able to walk stable and finish the gait sequences by using BW. Although the quadruped robot in this paper is able to secure the stability margin to walk, minimizations of landing impact and optimized control method for a gate generation are still remained problems. The solutions for these problems are now left for the future work. This work was supported in part by

the 2008 second Kyung-Hee University (KHU) graduate school excellent research paper scholarship.

References

1. Kumar, V., Waldron, K.J.: Force distribution in closed kinematic chains. IEEE Journal of Robotics and Automation 4(6), 657–664 (1988)
2. Hiroshi, I., Tamotsu, M., Fumio, H., Masayoshi, K.: Free gait for quadruped robots with posture control. In: 9th IEEE International workshop, pp. 433–438 (2006)
3. Fukuoka, Y., Kimura, H., Hada, Y., Takase, K.: Adaptive dynamic walking of a quadruped robot on irregular terrain using a neural system model. In: Proceeding of IEEE International Conference on Robotics and Automation, vol. 2, pp. 2037–2042 (2003)
4. Berkemeier, M.D., Sukthankar, P.: Self-organizing running in a quadruped robot model. In: 2005 IEEE International Conference on Robotics and Automation, pp. 4108–4113 (2005)
5. Takao, S., Gu, Z., Ikeda, T., Mita, T.: Realization of dynamic walking and running of a cat type quadruped robot using variable constraint control. In: 2003 SiCE Annual Conference, Fukui, pp. 3053–3058 (2003)
6. Yang, J.M.: Fault-tolerant gaits of quadruped robot for locked joint failures. IEEE Transactions on Systems, Man, and Cybernetics 32(4), 507–516 (2002)
7. Lee, Y.J., Sigoe, H.: Three-legged walking for fault tolerant locomotion of a quadruped robot with demining mission. In: IEEE/RSJ International Conference on Intelligent Robots and System, vol. 2, pp. 973–978 (2000)
8. Yang, J.M.: A fault tolerant gait for a hexapod robot over uneven terrain. IEEE Transactions on Systems, Man and Cybernetics 30(1), 172–180 (2000)
9. Shih, C.L., Klein, C.A.: An adaptive gait for legged walking machines over rough terrain. IEEE Transactions on System, Man and cybernetics 23(4), 1150–1155 (1993)

Development of a Micro Quad-Rotor UAV for Monitoring an Indoor Environment*

B.C. Min, C.H. Cho, K.M. Choi, and D.H. Kim

Department of Electrical Engineering, Kyung Hee University,
Seocheon-Dong, Giheung-Gu, Yongin-Si, 449-701, Korea
{minbc,movielove61,donghani}@khu.ac.kr
sm20s@naver.com

Abstract. The purpose of this paper is to develop a micro UAV (Unmanned Aerial Vehicle) as the platform for monitoring an indoor environment, more specifically, a quad-rotor aircraft that has strong advantages of its small size, stable hovering and precise flight. To begin with, we analyze the dynamics of a quad-rotor aircraft, and introduce control strategies based on the PD control. Then, the developed micro quad-rotor UAV, equipped with micro controllers, various sensors, and a wireless camera, is presented and is compared with the DraganFlyer commercial quad-rotor aircraft. Finally, experimental results are also provided so as to illustrate stable flight performances.

Keywords: UAV, quad-rotor aircraft, hovering, DraganFlyer.

1 Introduction

The UAV (Unmanned Aerial Vehicle) is widely used in many practical application e.g. research, surveillance and reconnaissance in specific regions or in dangerous regions which are inaccessible or hard to reach for other types of vehicles [1]. The UAV, which has such tremendous advantages, can be classified into two categories – fixed-wing aircraft and rotary-wing aircraft. When compared with fixed-wing aircrafts, rotary-wing aircrafts (rotorcrafts, helicopters) have specific characteristics like landing vertically in a limit space. Furthermore, such types are able to hover that make them suitable for applications that may be impossible to be performed using fixed-wing aircrafts [8].

Depending on their shape, rotary-wing aircrafts comprise several types of aircrafts. A single rotor aircraft, known as a standard helicopter, is one of the rotary-wing type aircrafts. This type aircraft is very hazardous to humans and other objects in an indoor environment due to the possibility that exposed rotor blades may collide with obstacles and cause the aircraft to crash. On the other hand, a quad-rotor aircraft is much safer in terms of the rotors, which are smaller and can be enclosed [3].

The most prevalent quad-rotor type UAV is DraganFlyer depicted as Fig.1, which is a radio-controlled quad-rotor aircraft commercially available from RCtoys[11]. Because of its tremendous advantages during flying, a number of research groups

* This work was supported in part by the Korea Science and Engineering Foundation (KOSEF) grant funded by the Korea government (MEST) (R01-2008-000-20352-0).

J.-H. Kim et al. (Eds.): FIRA 2009, LNCS 5744, pp. 262–271, 2009.

have used it as the platform in order to implement experiments on their research [1] [2]. However, some groups often complain an outrageous price and hardware, which is very large and heavy when used in indoor environments [10].

For all these reasons, we develop a platform with a low cost quad-rotor type UAV for monitoring an indoor environment by means of light frames, and compact circuits. To begin with, it is obvious that UAV should be equipped with stable hovering and precise flight: this is essential for an application like data acquisition using an on-board wireless camera for taking distinct images of the indoor environment. Accordingly, we have primarily focused on analyzing the dynamics of the quad-rotor aircraft and designing controllers for its stable flight.

The rest of the paper is organized as followed: Section 2 presents the dynamics of the quad-rotor aircraft. Section 3 introduces a PD controller for assuring a stable and optimal system. Section 4 presents a description of the platform that we have designed for the micro quad-rotor UAV and shows the summary results of the experimental operations of the quad-rotor UAV. Finally, Section 5 summarizes our finding and future scope of work to improve the capabilities of the UAV.

Fig. 1. DraganFlyer, from www.rctoys.com

2 Dynamic Modeling of the Quad-Rotor Aircraft

Deriving mathematical modeling or differential equations is necessary for the control of the quad-rotor position and altitude. However, it is hard for the complicated structure of the quad-rotor type to express its motion with only a simple modeling. In addition, since the quad-rotor type aircraft includes highly nonlinear factors, we need to consider several assumptions in order to get a desired model [6].

1) The body is rigid and symmetrical.
2) The rotors are rigid, i.e. no blade flapping occurs.
3) The difference of gravity by altitude or the spin of the earth is minor.
4) The center of mass and body fixed frame origin coincide.

These assumptions can be formed because of slower speed and lower altitude of the quad-rotor aircraft whose body is rigid having 6 DOF (Degree of Freedom) as compared to a regular aircraft. Under these assumptions, it is possible to describe the fuselage dynamics. Therefore, to mathematically illustrate the fuselage dynamics of the quad-rotor type aircraft, a coordinate system should be defined. The coordinate system can be divided into an earth frame {E} and a body frame {B} as shown in Fig. 2.

The rotational transformation matrix between the earth frame and the body frame can be obtained based on Euler angles in Fig. 2 [1].

$$R_{EB} = R_\psi R_\theta R_\phi = \begin{bmatrix} C_\theta C_\psi & C_\psi S_\theta S_\phi - C_\phi S_\psi & C_\phi S_\theta C_\psi + S_\phi S_\psi \\ C_\theta S_\psi & S_\theta S_\phi S_\psi - C_\phi C_\psi & C_\phi S_\theta S_\psi - C_\psi S_\phi \\ -S_\theta & C_\theta S_\phi & C_\theta C_\phi \end{bmatrix} \tag{1}$$

where C and S indicate the trigonometric cosine and sine functions, respectively.

The transformation of velocities between the earth frame and body frame can be derived from Eq. (1).

$$\begin{bmatrix} \upsilon \\ v \\ \omega \end{bmatrix} = R_{EB} \begin{bmatrix} \upsilon_B \\ v_B \\ \omega_B \end{bmatrix} \tag{2}$$

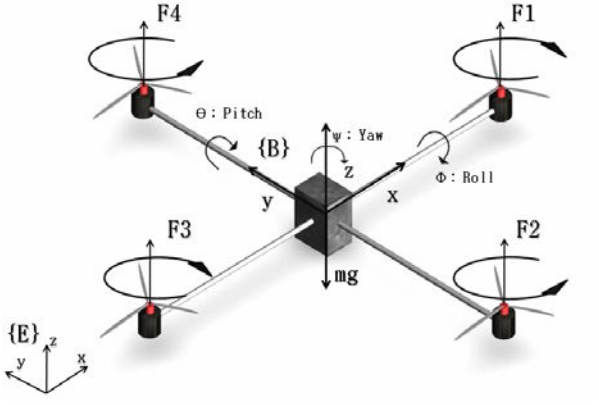


Fig. 2. The coordinate system with an earth frame {E} and a body frame {B}

Similarly, positions, forces, moments, accelerations and rotational velocities can be transformed based on R_{EB} between coordinate systems. In the body frame, the forces are presented as

$$F_B = \begin{bmatrix} F_{xB} \\ F_{yB} \\ F_{zB} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} F_1 \\ F_2 \\ F_3 \\ F_4 \end{bmatrix} \tag{3}$$

Accordingly, in the earth frame, the forces can be defined as

$$F_E = \begin{bmatrix} F_{xE} \\ F_{yE} \\ F_{zE} \end{bmatrix} = R_{EB} F_B = \begin{bmatrix} C_\phi C_\psi S_\theta + S_\phi S_\psi \\ C_\phi S_\theta S_\psi - C_\psi S_\phi \\ C_\phi C_\theta \end{bmatrix} (\sum_{i=1}^{4} F_i) \tag{4}$$

Therefore, equations of motion in the earth frame are derived by the Newton's laws.

$$m \begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} = \begin{bmatrix} F_{xE} \\ F_{yE} \\ F_{zE} \end{bmatrix} - \begin{bmatrix} K_{fx} & 0 & 0 \\ 0 & K_{fy} & 0 \\ 0 & 0 & K_{fz} \end{bmatrix} \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} - \begin{bmatrix} 0 \\ 0 \\ mg \end{bmatrix} = \begin{bmatrix} F_{xE} - K_{fx}\dot{x} \\ F_{yE} - K_{fy}\dot{y} \\ F_{zE} - K_{fz}\dot{y} - mg \end{bmatrix} \tag{5}$$

where K_{fx}, K_{fy}, K_{fz} are the drag coefficients, which can be ignored at low speeds.

By extending the Newton's law on rotational dynamics, its equation can be written as [1]

$$\ddot{\theta} = \frac{l(K_l(\omega_3^2 - \omega_1^2))}{I_y} \tag{6}$$

$$\ddot{\phi} = \frac{l(K_l(\omega_4^2 - \omega_2^2))}{I_x} \tag{7}$$

$$\ddot{\psi} = \frac{K_d(\omega_1^2 + \omega_3^2 - \omega_2^2 - \omega_4^2)}{I_z} \tag{8}$$

where l is the distance from the center of rotors to the quad-rotor aircraft center of mass. K_l and K_d indicate the lift coefficient and drag coefficient respectively and w_i is the angular rotor speed. I_x, I_y, I_z are moments of inertia of the quad-rotor.

As a consequence, the complete dynamic model which governs the quad-rotor aircraft is as follows:

$$\begin{cases} \ddot{x} = \dfrac{(\cos\phi\sin\theta\cos\psi + \sin\phi\sin\psi)u_1 - K_{fx}\dot{x}}{m} \\ \ddot{y} = \dfrac{(\cos\phi\sin\theta\sin\psi - \sin\phi\cos\psi)u_1 - K_{fy}\dot{y}}{m} \\ \ddot{z} = \dfrac{(\cos\phi\cos\theta)u_1 - K_{fz}\dot{z}}{m} - g \\ \ddot{\phi} = \dfrac{l(u_2 - K_l\dot{\phi})}{I_x} \\ \ddot{\theta} = \dfrac{l(u_3 - K_l\dot{\theta})}{I_y} \\ \ddot{\psi} = \dfrac{(u_4 - K_d\dot{\psi})}{I_z} \end{cases} \tag{9}$$

where u_i (i=1, 2, 3, 4) are control inputs of the model.

$$\begin{cases} u_1 = F_1 + F_2 + F_3 + F_4 \\ u_2 = F_4 - F_2 \\ u_3 = F_3 - F_1 \\ u_4 = F_1 + F_3 - F_2 - F_4 \end{cases} \tag{10}$$

3 Control of the Quad-Rotor Aircraft

The PD controllers, which are widely used in the process industries, are implemented on each of the four axes of the quad-rotor aircraft, namely altitude, roll, pitch and yaw as depicted in Fig. 3 [5].

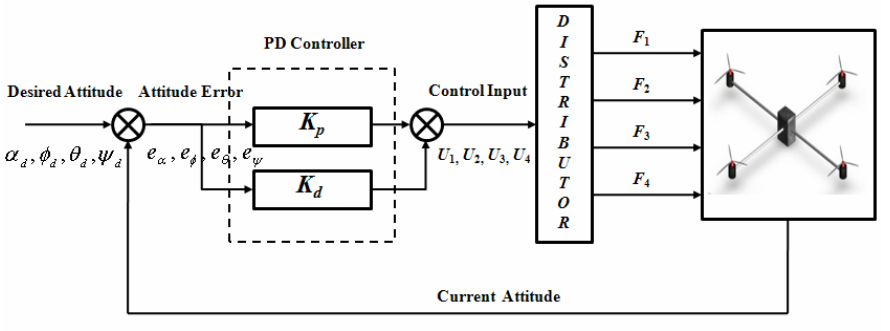


Fig. 3. A structure of the PD controller for controlling the altitude and the attitude of the micro quad-rotor UAV

3.1 Altitude Control

U_1, which is the control input with respect to the desired altitude a_d, is obtained from the PD controller [2] [7].

$$U_1 = \frac{K_{p1}(\alpha_d - \alpha) + K_{i1}\int_0^t (\alpha_d - \alpha)_{d\tau} + K_{d1}(\dot{\alpha}_d - \dot{\alpha}) + mg}{\cos\theta\cos\phi} \tag{11}$$

In Eq. (11), K_{p1}, K_{i1} and K_{d1} indicate a proportional, integral, and differential positive gains respectively. However, since effects causing a steady-state error like wind are not considered in this paper, integral terms can be eliminated. Hence, Eq. (11) can be simplified as

$$U_1 = \frac{K_{p1}(\alpha_d - \alpha) + K_{d1}(\dot{\alpha}_d - \dot{\alpha}) + mg}{\cos\theta\cos\phi} \tag{12}$$

3.2 Attitude Control

The motion along the x-axis and y-axis are related to the pitch and roll angles, respectively [4]. Hence, U_2 is given by

$$U_2 = K_{p2}(\phi_d - \phi) + K_{d2}(\dot{\phi}_d - \dot{\phi}) \tag{13}$$

The desired roll angle, in terms of the error between the desired angle and the actual angle, can be written as

$$\phi_d = \arcsin(K_p(y - y_d) + k_d(\dot{y} - \dot{y}_d)) \tag{14}$$

Similarly, the control input U_3, the desired pitch angle and the desired pitch angle velocity can be written as follows:

$$U_3 = K_{p3}(\theta_d - \theta) + K_{d3}(\dot{\theta}_d - \dot{\theta}) \tag{15}$$

$$\theta_d = \arcsin(K_p(x - x_d) + K_d(\dot{x} - \dot{x}_d)) \tag{16}$$

Furthermore, the control input U_4 for the control of the yaw can be derived as

$$U_4 = K_{p4}(\psi_d - \psi) + K_{d4}(\dot{\psi}_d - \dot{\psi}) \tag{17}$$

4 Experimental Setup and Results

4.1 Experimental Setup

Main board. The main board architecture of the developed micro quad-rotor UAV is shown in Fig. 4 (a). The developed quad-rotor UAV has a 8-bit micro controller (Atmega88V) which is in charge of managing sensors, controlling four dc geared motors and communicating with a controller by a wireless. It is thus equipped with compact on-board sensors composed of two gyro modules, an accelerometer for attitude control of the UAV and a magnetic compass sensor for the control of its heading directions. Moreover, to receive the command data from the wireless controller (Fig. 4 (b)), a zigbee module allowing the wireless communication via a serial port, is also composed.

Hardware. To reduce the weight of the platform, carbon fiber robs were used for the whole body as depicted in Fig 4 (c). In addition, we have tried to keep the micro quad-rotor UAV as small as possible so that it performs well in an indoor environment. As a result, its specification, briefly given in Table 1, strikingly differs with the DraganFlyer. In Table 1, the lighter total weight of the developed quad-rotor UAV with batteries, which is almost one tenth of the DraganFlyer, means that it can be an efficient system compared with the DraganFlyer in terms of energy. Furthermore, the smaller size, which is a half of the DraganFlyer, signifies that it will be helpful to utilize it in an indoor environment.

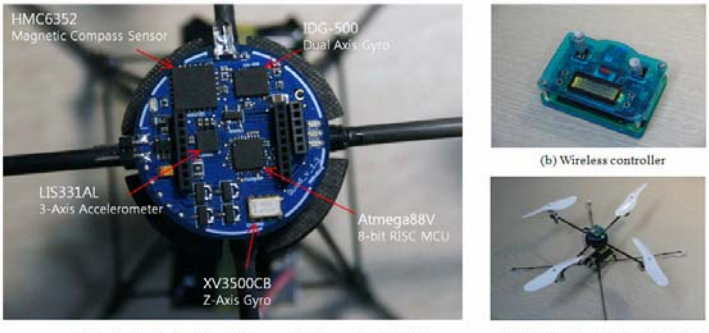


Fig. 4. The scheme of the developed micro quad-rotor UAV system

Table 1. Comparisons between the developed micro quad-rotor UAV and the DraganFlyer

Parameter	Developed micro quad–rotor UAV	DraganFlyer
total weight	41 g	482 g
height	85 mm	110 mm
diameter	300 mm	760 mm
blade	4″	12.375″

Table 2. Parameter values used in PD controllers

Parameter	Value	Parameter	Value
K_{p1}	0.32	K_{d1}	2.21
K_{p2}	7	K_{d2}	0.8
K_{p3}	17	K_{d3}	4.42
K_{p4}	1.23	K_{d4}	0.73

4.2 Experiments and Results

Fig. 5 shows that measured angle data from the accelerometer sensor, which is attached on the circuit board, are mixed with noise when speeding up motors. To eliminate the noise arising from vibrations of the body, we designed the RC filter composed with several resistors and capacitors.

The performance of the proposed control approach is illustrated in Fig. 6 through 8. Fig. 6 and Fig. 7 show the response of the PD controllers to stabilize the micro quad-rotor UAV, which was implemented from the tilted body at 40 degrees on the test bench as depicted in Fig. 9 on the lower right. In addition, we ignored any disturbances like wind during experiments because we assumed that the developed UAV is only used indoors.

Parameters that have been used for PD controllers are given in Table 2. Most parameters were derived by tuning gains until obtaining a better response of the system on the test bench [9]. However, PD gains of yaw control were nakedly obtained without the test bench because the test bench only allows UAV to roll from side to side.

In Fig. 6 and Fig. 7, left columns display the late response of roll and pitch angle with inaccuracy parameters, and right columns present the faster response of roll and pitch angle with adequate parameters. Thus, Fig. 8 shows the roll and pitch angles during hovering indoors. Accordingly, the rotational subsystem of the micro quad-rotor UAV seems to be completely controllable and observable. Finally, Fig. 9 shows an indoor flying test of the developed micro quad-rotor UAV with the designed wireless controller depicted in Fig. 4(b). Although it was not fully autonomous flight, we verified the effectiveness of its performance in the real indoor environments.

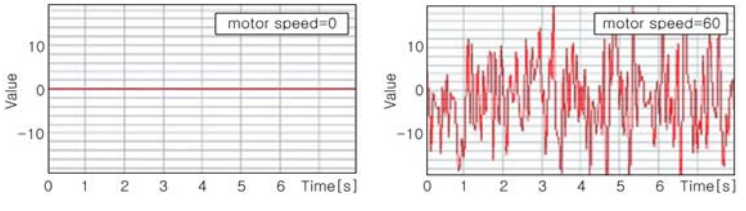


Fig. 5. Measured sensor data with noise when speeding up motors

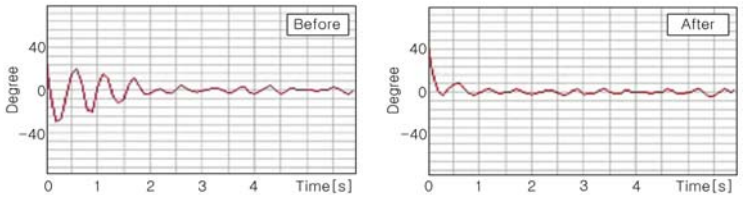


Fig. 6. Variations of roll angles with PD control

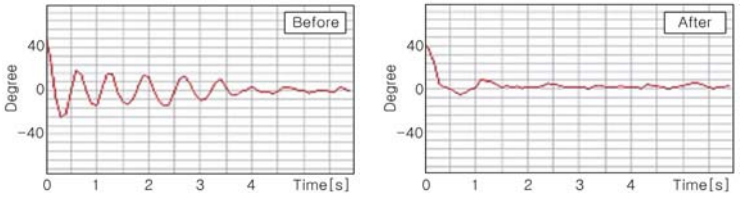


Fig. 7. Variations of pitch angles with PD control

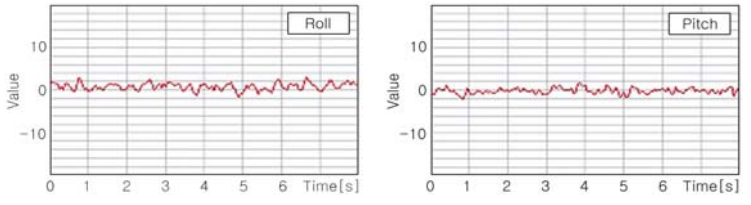


Fig. 8. Experimental results of the quad-rotor at hover

Fig. 9. The figure on the lower right is the designed test bench for experiments, and the figure on the center is the developed micro quad-rotor UAV during flight indoors

5 Conclusion

Since DraganFlyer, which is commercially the quad-rotor type UAV, has tremendous advantages as the platform, a number of research groups have used it so as to implement experiments on their research. However, some groups often complain that it is outrageous price and large size.

Accordingly, in this paper, we have presented the platform composed of the developed the micro quad-rotor UAV and the designed stabilization control algorithm. The developed quad-rotor UAV was built with light frames, compact circuits as small as possible. Furthermore, it was implemented with low cost, within the premise that its specification is outstanding compared with DraganFlyer.

As a result, the experimental results have shown that the designed platform is performing well in terms of stable flight. However, we did not verify the control of the altitude because the developed quad-rotor UAV is not yet equipped with an ultrasonic sensor measuring the distance between the ground and it even though the control strategy is designed in this paper. Hence, we will use various sensors in future research in order to improve the flight performance, and further study will be focused on autonomous flight. To be specific, a navigation system and the control strategies for avoiding obstacles will be studied.

References

1. Castillo, P., Lozano, R., Druz, A.: Modelling and Control of Mini-Flying Machines, 1st edn. Springer, London (2005)
2. Lara, D., Sanchez, A., Lozano, R., Castillo, P.: Real-Time Embedded Control System for VTOL Aircrafts: Application to stabilize a quad-rotor helicopter. In: 2006 IEEE Symposium Intelligent Control, pp. 2553–2558 (2006)
3. Valavanis, K.P.: Advances in Unmanned Aerial Vehicles, 1st edn. Springer, Heidelberg (2007)
4. Erginer, B., Altuġ, E.: Modeling and PD Control of a Quadrotor VTOL Vehicle. In: IEEE explore of Intelligent Vehicles Symposium, pp. 894–899 (2007)
5. Park, S., Won, D.H., Kang, M.S., Kim, T.J., Lee, H.G.: RIC (Robust Internal-loop Compensator) based flight control of a quad-rotor type UAV. In: 2005 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 1015–1020 (2005)
6. Bouadi, H., Bouchoucha, M., Tadjine, M.: Sliding Mode Control based on Backstepping Approach for an UAV Type-Quadrotor. International Journal of Applied Mathematics and Computer Sciences 4(1), 12–17 (2008)
7. Kuo, B.C., Golnaraghi, F.: Automatic Control Systems, 8th edn. John Wiley & Sons. Inc., Chichester (2003)
8. Bouabdallah, S., Murrieri, P., Siegwart, R.: Design and control of an indoor micro quadrotor. In: Proceedings of IEEE International Conference on Robotics and Automation, vol. 5, pp. 4393–4398 (2004)
9. Ledin, J.: Simulation Engineering, 1st edn. CMP Books (2001)
10. Lillywhite, K., Lee, D.J., Tippetts, B., Fowers, S., Dennis, A., Nelson, B., Archibald, J.: An embedded vision system for an unmanned four-rotor helicopter. Robotic Vision Laboratory, Department of Electrical and Computer Engineering, BrighamYoung University
11. DraganFlyer, `http://www.rctoys.com/draganflyer3.php`

An Improved Method for Real-Time 3D Construction of DTM

Yi Wei

Polytechnic Institute of New York University
yiwei1985@gmail.com

Abstract. This paper discusses the real-time optimal construction of DTM by two measures. One is to improve coordinate transformation of discrete points acquired from lidar, after processing a total number of 10000 data points, the formula calculation for transformation costs 0.810s, while the table look-up method for transformation costs 0.188s, indicating that the latter is superior to the former. The other one is to adjust the density of the point cloud acquired from lidar, the certain amount of the data points are used for 3D construction in proper proportion in order to meet different needs for 3D imaging, and ultimately increase efficiency of DTM construction while saving system resources.

Keywords: 3D Model, Delaunay, Real-Time, Terrain.

1 Introduction

A digital terrain model (DTM) can be defined as a numerical model of a terrain surface, including the attributes of texture, altitude and other related information which makes it possible to calculate the elevation of any point within the domain of the model. It can be generated by computer systems and has great potential in the field of terrain detection, geographic exploration and remote sensing imaging technology.

In practice, how to ensure the real-time display of DTM is of great significance. The process has high demands to the computer system, so how efficiently to generate the model of terrain information often becomes a valuable topic for research even though 3D modeling technology has well developed in recent years and is relatively mature. However, in certain circumstances, such as when the researchers only need to know the contour of a specific scanned area, and ignore the detailed elements on the terrain surface, it is important to know how to properly reduce the corresponding overhead for processing in the computer imaging system [4]. To solve this problem, this paper has a preliminary exploration to the real-time Three-Dimensional construction of DTM.

2 Principle

The terrain information is collected by the 3D imaging system in the form of discrete points which come from acquisition card [15] read by interrupt interface program. Then, the constrained Delaunay triangulation starts to connect

J.-H. Kim et al. (Eds.): FIRA 2009, LNCS 5744, pp. 272–281, 2009.

the acquired topographic data points for generating spatial tetrahedron and completing the mathematical modeling. Terrain information will eventually be transformed into 3D model (Fig.1).

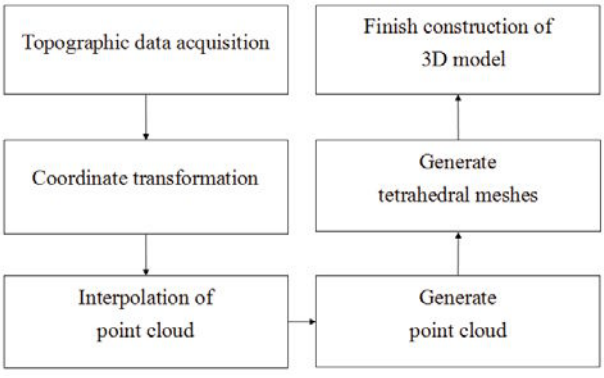


Fig. 1. System flowchart of 3D construction of DTM

2.1 Acquisition of Terrain Data

In this paper, all the obtained DTM data are from lidar scanning system (Fig.2) [16].

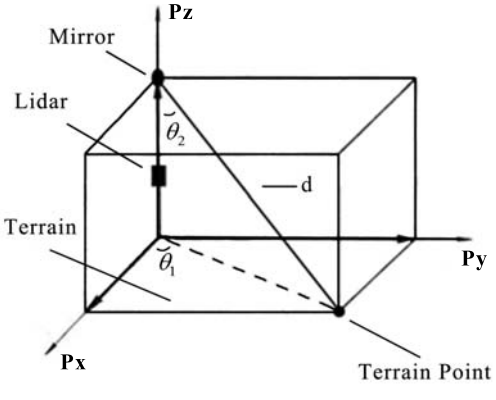


Fig. 2. Work principle of data acquisition system

According to the definition of coordinates transformation [11], given the sample collection $[p_1, p_2, p_3, \ldots, p_{n-1}, p_n]$ consisting of data points$(\theta_{n1}, \theta_{n2}, d_n)$, the

relationship between the two adjacent sampling points can be expressed as a coordinate formula (1).

$$\begin{aligned}&[(\theta_{11},\theta_{12},d_1),(\theta_{21},\theta_{22},d_2),\ldots,(\theta_{(n-1)1},\theta_{(n-1)2},d_{(n-1)}),(\theta_{n1},\theta_{n2},d_n)]\\&=[(P_{x1},P_{y1},P_{z1}),(P_{x2},P_{y2},P_{z2}),\ldots,\\&(P_{x(n-1)},P_{y(n-1)},P_{z(n-1)}),(P_{xn},P_{yn},P_{zn})]\end{aligned}\tag{1}$$

According to work principle of data acquisition system (Fig.2), transformation formula (2) of collected lidar data to DTM data can be derived.

$$\begin{aligned}P_x&=d\times\sin(\theta_2)\times\cos(\theta_1)\\P_y&=d\times\sin(\theta_2)\times\sin(\theta_1)\\P_z&=d\times\cos(\theta_2)\end{aligned}\tag{2}$$

2.2 Constrained Delaunay Triangulation

Using LOP algorithm to optimize process of Delaunay triangulation is the key step to each triangle in point cloud to construct 3D model [9][12]. The basic process is that the new generated triangles share the same diagonal with the adjacent triangle forming a quadrilateral (Fig.3), in which triangles should be examined one by one to make sure whether they obey Empty Circumcircle Principle (ECP) and Local Equiangularity Principle (LEP) or not [8][14]. If the ECP is not satisfied, then triangles will be regenerated with new points that are fallen into the existing circumcircle. If LEP is not satisfied, then swap adjacent triangles that belong to the same diagonal consisting in their quadrilateral. The two approaches can be applied continuously until all the triangles satisfy the principle. The whole process is carried out with the successive insertion of each point for forming point cloud, and its algorithm is achieved via recursive invocation [6].

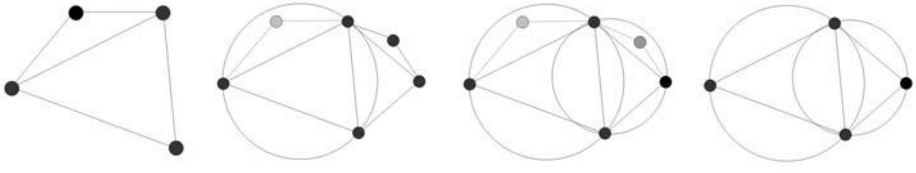
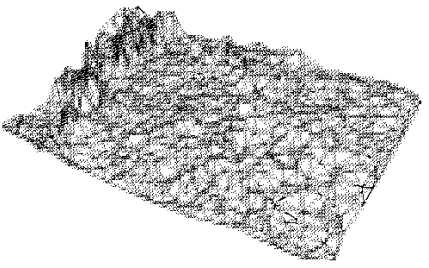


Fig. 3. Construction of constrained Delaunay triangulation

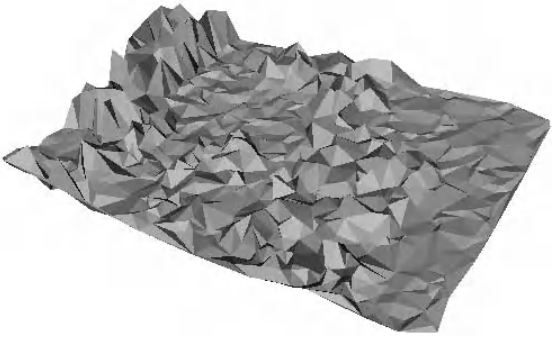
After the transformation, the obtained data of rectangular coordinates were drawn to generate the discrete points, which constitute point cloud (Fig.4(a)).

The framework of DTM is constructed from point cloud using constrained Delaunay triangulation [1][3], which optimizes data transformation by ECP and LEP [6], and finally finishes the construction of tetrahedral meshes of terrain model (Fig.4(b)).

(a) Model of terrain point cloud



(b) Model of terrain tetrahedral meshes



(c) Model of terrain profile

Fig. 4. Construction of model

Then, dealing with the coordinate points on the basis of tetrahedral terrain can display the model of terrain profile (Fig.4(c)) so as to build a solid foundation for establishing the following real-time 3D terrain model.

3 Improved Method

Real-time DTM display is a challenging task with great significance as the process will demand much resource of the computer system, so how fast to generate contour model of terrain by scanning often becomes key part of an assessment of the scientific and technical features. In view of this, the present article attempts to take measures to optimize the coordinate transformation and data acquisition of discrete terrain points.

3.1 Optimization of Coordinates Transformation for Real-Time Dtm Construction

Coordinate transformation of data points is the basis to guarantee the correctness of DTM, but a good method of coordinate transformation, can not only guarantee the accuracy of the data transformation, but take increasing efficiency of which into consideration. According to previous derived formula (2), we can

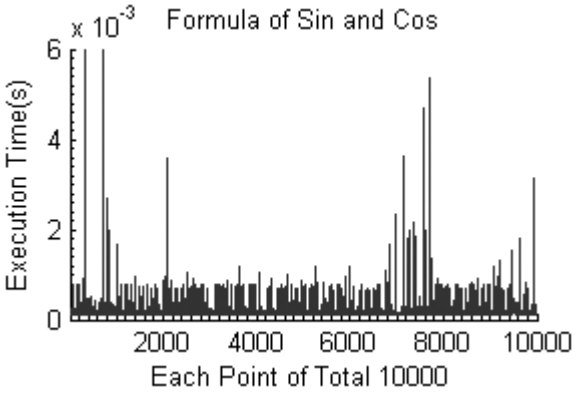


Fig. 5. Execution time of transformation by formula

know that because the calculation of sine and cosine in code spends too much time (Fig.5), that is to say, the speed of direct transformation is slow for the coordinate point data P(x, y, z) in accordance with the formula calculation.

Studies have shown that the table look-up method can effectively improve the efficiency of coordinate transformation that always faced as a problem. This method is also widely used in the sensor on data processing and other fields [5]. According to the periodic characteristics of sine and cosine, it is easy to find out the rule of continuous collected angles for imaging. So, as long as those data are stored in advance, each mapping can be setup between them. Transformations between polar coordinates$(\theta_{n1}, \theta_{n2}, d_n)$ and rectangular coordinates(P_{xn}, P_{yn}, P_{zn})need to make comparison for value-pair adopting dynamic table lookup [2], which can save more time for calculations of sine and cosine (Fig.6).

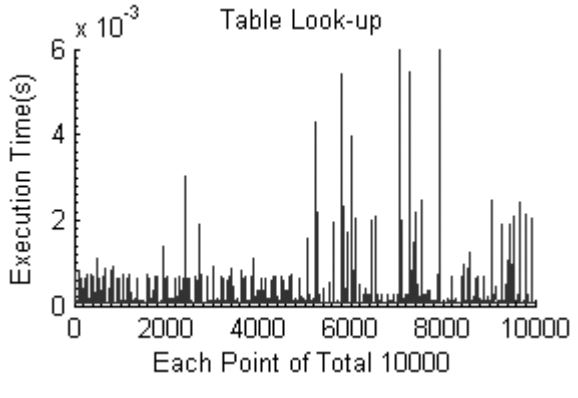


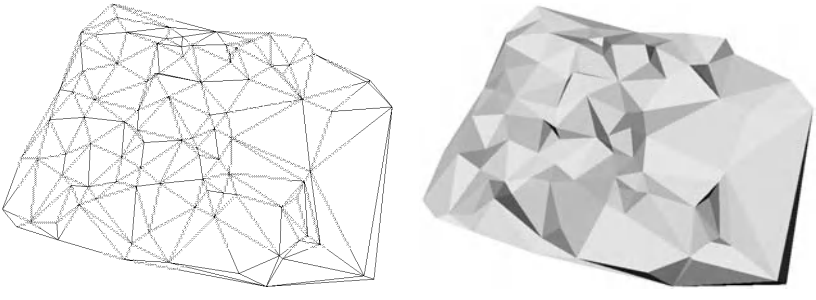
Fig. 6. Execution time of transformation by table look-up

In summary, it is a better choice to apply table look-up for data transformation, rather than dealing with sine, cosine computing. This paper basically attempts to achieve rapid sampling of points to improve the efficiency of data transformation.

3.2 Optimization of Data Point Acquisition for Real-Time DTM Construction

Although it is important for research to entirely display and analysis the real topography 3D model, sometimes in practical applications, needs should be met in specific situations, for example, the user only wants to acquaint the roughly contour and profile of specific topographic regions without displaying much details [7]. By means of omitting detailed terrain points for simulation, the imaging process can be speeded up so that users may obtain the necessary information of terrain model quickly.

Through adjusting the unit area of the discrete points, the system overhead of dealing with graphic data can be reduced [10]. For comparison purposes, a total of collected 10000 terrain points in $100m^2$ areas construct the meshes and the profiles of terrain model adopting respectively 1%, 10%, 50% and 100% of its total number of collected points. The result shows that adopting 1% of the total data points, the overhead of imaging is little and can be neglected, but the terrain model seems very rough visually (Fig.7).



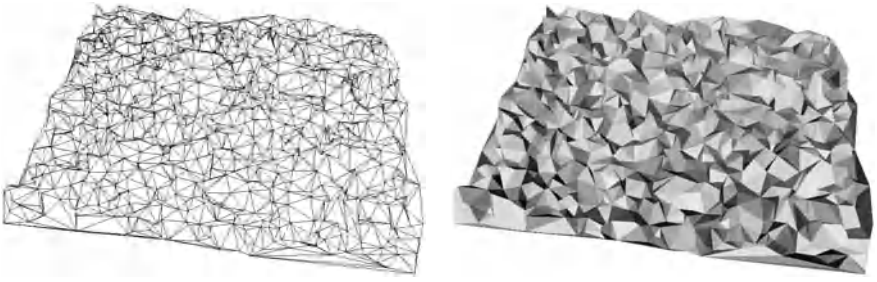
(a) Mesh model (b) Profile model

Fig. 7. 3D Imaging effect of 1% data points

While comparing to 1% one, the effect of adopting 10% of data points for imaging has been slightly improved (Fig.8).

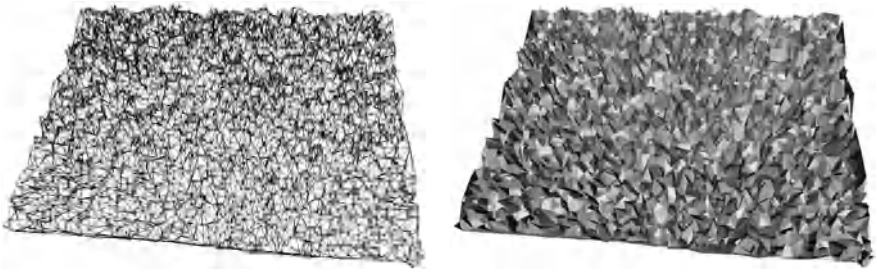
The display effect adopting 50% of the data points has been enhanced (Fig.9). However, with the increase of number of data points, imaging time will be expanded.

When all of the data points 100% of which displayed on the screen (Fig.10), the broken ground situations are distinct. Obviously, if we only need to know the profile of it, this will increase the difficulty of topography analysis, and the



(a) Mesh model (b) Profile model

Fig. 8. 3D Imaging effect of 10% data points



(a) Mesh model (b) Profile model

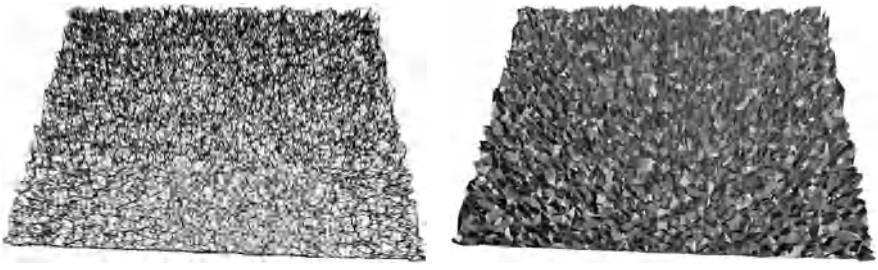
Fig. 9. 3D Imaging effect of 50% data points

process of 3D model construction cannot be tolerated in terms of time, speed and system overhead.

By calculating and analyzing, when no density control to point cloud of underlying terrain construction, the average execution time for imaging from entire points to its meshes then to its profiles is 4.397s (Fig.11(a)). It can be seen from this that using the approach mentioned to construct model will have a relatively heavy overhead to real-time system.

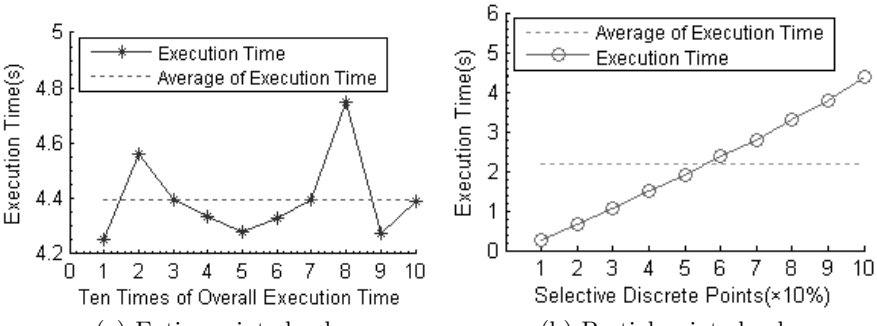
After adjusting density control to terrain data, the average time of imaging is 2.198s. The number of data points will have a conspicuous impact to the running speed of system. Research indicates that the optimal value can be taken from the average time of entire points' construction (Fig.11(a)) to average time of partial points' construction (Fig.11(b)). This is a tradeoff between effect and efficiency of model construction which will not only minimize losses of the terrain details, but maximize the efficiency of the construction of terrain model.

The method adopted for terrain simulation, can generate spatial point cloud according to different needs for terrain model construction, under the premise to guarantee a distinct profile of 3D modeling (Fig.11(b)). A similar approach of this has been tried on Morphometric analysis and model construction [13]. When dealing with more discrete data, this method has superiority obviously.



(a) Mesh model (b) Profile model

Fig. 10. 3D Imaging effect of 100% data points



(a) Entire point cloud (b) Partial point cloud

Fig. 11. Modeling execution time

The result shows that the computing process of imaging has saved about at least 50% of system resources, and reduced the imaging delay caused by data processing overhead in order to prevent system from being freezing or crashed.

4 Conclusion

This paper discusses the real-time optimal construction of DTM. By fast coordinate transformation and improvement of data acquisition, the construction speed of 3D terrain model can be increased significantly, and the demonstrated effect of model can be guaranteed. Through analyzing average density of point cloud in different proportion, terrain discrete points will be triangulated based on Delaunay triangulation with LOP algorithm. Ultimately the data can be transformed into spatial tetrahedron, which has completed the transformation process from DTM to 3D model.

Using table look-up method to carry out the transformation from the rectangular coordinates to polar coordinates can decrease the delay of 3D terrain imaging in real-time scanning process. Under the condition of using 10000 points for testing purpose, the execution time of formula sine and cosine costs 0.810s,

under the same condition, the execution time of table look-up method costs 0.188s, which has saved 0.622s. The result shows that table look-up method has higher efficiency then the former one on coordinate transformation.

Before 3D model being constructed successfully, the imaging processing time can be shortened as much as possible by reducing the number of discrete points in terrain scene. In the specific application, algorithms are used to determine how to filter data points for further use or discarding. Without prejudice to the 3D terrain profile, non-critical data points should be reduced in average density appropriately to ensure construction of relatively accurate model [13].

The improved real-time displaying of DTM is not only used to analyze the terrain target quickly, but also to give feedback information to systems and users in time for decision making. It is of particular significance under certain circumstances, when the data acquisition and 3D imaging system are integrated as sub-system into the entire framework of system. It also provides prerequisite conditions for feedback control and reduces the feedback delay, which can be applied in the field of automatic routing, automatic avoidance of obstacles and etc.

When dealing with a small number of points, though terrain conditions cannot reflect too many details, the overhead of computer system is relatively light. When dealing with a large number of points, more complicated 3D models can be generated, but more system overhead will have to affect the speed of imaging. In the stage of 3D model construction, through using table look-up coordinate transformation and density control to point cloud scanned from unit area, the efficiency of real-time construction of DTM has been improved remarkably.

References

1. Boissonnat, J.D., Faugeraus, O.D., Le Bras-Mehlmam, E.: Representing Stereo Data with the Delaunay Triangulation. Artificial Intelligence 44, 41–87 (1990)
2. Clarke, N., Cantoni, A.: Implementation of Dynamic Look-up Tables. Computers and Digital Techniques 141, 391–397 (1994)
3. Martinoni, D., Bernhard, L.: A Conceptual Framework for Reliable Digital Terrain Modeling. In: Spatial Data Handling Conference, pp. 737–750 (1998)
4. Dupont, F., Deseilligny, M.P., Gondran, M.: DTM Extraction from Topographic Maps. In: Document Analysis and Recognition Conference, pp. 475–478 (1999)
5. Flammini, A., Marioliz, D., Taroni, A.: Application of an Optimal Look-up Table to Sensor Data Processing. Instrumentation and Measurement 48, 813–816 (1999)
6. Li, G., Hao, Y.-l., Zu, W.: A Modifiedconstrained Delaunay Triangulation Algorithm Based on Extracted Boundary Characteristic Points. In: International Conference on Mechatronics and Automation, pp. 873–878 (2007)
7. Guedes, L.C.C.: Real-time Terrain Surface Extraction at Variable Resolution. In: Computer Graphics and Image Processing, pp. 87–94 (1997)
8. O'Rourke, J. (ed.): Computational Geometry in C, pp. 161–173. Cambridge University Press, Cambridge (1997)
9. Kong, J.-H., Zheng, J.-B.: An Algorithm of Generating Unstructured Tetrahedron from 3D Discrete Points. Machine Learning and Cybernetics 5, 2770–2774 (2008)

10. Kakarlapudi, S., Uijt de Haag, M.: The Application of Image Analysis Techniques to Forward Looking Terrain Database Integrity Monitoring. In: Digital Avionics Systems Conference, vol. 1, pp. 4–6 (2004)
11. Rader, C.M.: Generating Rectangular Coordinates in Polar Coordinate Order. Signal Processing Magazine 22, 178–180 (2005)
12. Rognant, L., Chassery, J.M., Goze, S., Planes, J.G.: The Delaunay Constrained Triangulation: The Delaunay Stable Algorithms. Information Visualization, 147–152 (1999)
13. Shirkhodaie, A., Amrani, R., Tunstel, E.: Soft Computing for Visual Terrain Perception and Traversability Assessment by Planetary Robotic Systems. In: Systems, Man and Cybernetics Conference, vol. 2, pp. 1848–1855 (2005)
14. Wang, C.A., Schubert, L.: An Optimal Algorithm for Constructing the Delaunay Triangulation of a Set of Line Segments. In: Proceedings of the 3rd Annual ACM Symposium on Computational Geometry, pp. 223–232 (1987)
15. Leow, W.K., Huang, Z., Zhang, Y., Setiono, R.: Rapid 3D Model Acquisition from Images of Small Objects. In: Geometric Modeling and Processing, pp. 33–41 (2000)
16. Wei, Y.: High-speed Lidar Data Processing Technology. Bachelor Dissertation of Beijing University of Technology, pp. 12–20 (2007)

Depth Estimation Using Variant of Depth of Field by Horizontal Planes of Sharp Focus

Hiroshi Ikeoka and Takayuki Hamamoto

Department of Electrical Engineering, Faculty of Engineering
Tokyo University of Science
Tokyo, Japan
{ikeoka,hamamoto}@isl.ee.kagu.tus.ac.jp

Abstract. We have been investigating depth estimation techniques using variant of depth of field (DOF) by tilted optics imaging. These techniques can be applied for robotic and automation tasks, because it needs extremely fewer multiple focus images for depth estimation than the conventional passive methods. Hence, our method gets rid of the bottleneck of passive methods with a single camera; the motion speed of optical mechanics is very slower than that of image processing parts. Therefore, it is suitable for the high speed depth estimation like real-time processing accomplished by hardware such as a smart image sensor or an FPGA.

Keywords: Depth, Distance, Depth of field, Tilted optics.

1 Introduction

Computer vision technologies have advanced rapidly and considerable research has been conducted on the development of practical applications using technological advancements such as those in high-performance image sensors. In these technologies, depth estimation is considered a very important research field. There exist two types of depth estimation methods that use image sensing with a single camera: the "active" method employs the projection and reflection of artificial light such as infrared and laser light from the imaging device, and the "passive" method uses the reflection of environment light such as sunlight. In general, the composition of the latter is simpler than that of the former.

Irrespective of which method is chosen, it is necessary for the depth estimation method to deal with a large amount of comparisons and operations; therefore, the achievement of a high speed performance is a major challenge. A probable solution is the use of hardware on which the depth estimation function is implemented. In particular, some research groups have successfully used high performance hardware based on the active method [1][2].

On the contrary, we adopt the passive method that possesses a simpler composition. The passive depth estimation methods comprise the "depth from focus" (DFF) technique and the "depth from defocus" (DFD) technique [3][4][5]. The depth

J.-H. Kim et al. (Eds.): FIRA 2009, LNCS 5744, pp. 282–291, 2009.

estimation method using the DFF technique is achieved by finding for the most in-focus position over multiple focus images [3][6]. In fact, we have investigated the high speed depth estimation system based on DFF, and developed a smart image sensor that performs both edge detection and in-focus judgment, and have proposed an interpolation method to determine the depth in textureless regions [6][7]. These studies have contributed to improving the speed of depth estimation; however, the speed at which multiple focus images can be captured is considerably slower than that at which other processing steps can be completed. Thus, to avoid this bottleneck, we propose a new method that uses only a few multiple focus images.

In the following sections, we first indicate the drawbacks of the conventional method. Then, we describe the tilted optics and the variant of DOF of that; these are used in our method. We then explain our depth estimation method by the variant of DOF. Finally, we present a few experimental results to demonstrate the performance of the proposed method.

2 The Drawback of Conventional Method with a Single Camera for Real-Time Usage

The conventional method requires many "Planes of sharp focus" (POF) that are confronted in front of the image sensor. Generally, as shown in Fig.1 we should obtain a number of multiple focus images; 1) by repeatedly changing the focal position by a small amount using a mechanical component, or 2) by spectroscopy optics that has limit of division number. As a consequence, there is a trade-off between the estimation speed and the estimation accuracy. Therefore, this process is a bottleneck for high-speed depth estimation using the conventional passive method with a single camera like the DFF and DFD techniques. And without modification, the conventional method is considered unsuitable for real-time depth estimation. It is disadvantage for developing a practical depth estimation system.

To solve these problems, as we estimate the depth value by changing the arrangement of the POFs such that they are horizontal. We explain the details of depth estimation in the following sections.

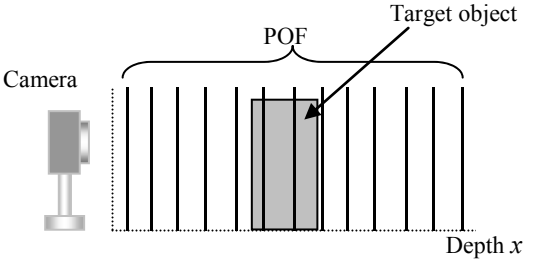


Fig. 1. Conventional methods (DFF, DFD)

3 Depth Estimation by the Variant of DOF

3.1 A Titlted POF

Now let us consider the situation depicted in Fig. 2: an image sensor is placed on the left side of horizontal axis and an optical unit (lens) which has the tilted angle, θ, with respect to the image sensor. Similarly, a POF is tilted like Fig. 2; it is different from normal optics. (L_x, L_y) is a center of rotation for the lens, and H is a distance between a first principal point and the center of rotation, H' is between a second principal point and the center of rotation. There are two light lines that reach a POF via each focal point, and a light line that reaches a POF via two principal points. Hence, we can get an intersecting point of these light lines on a POF; x and y coordinates are expressed with Y on the image sensor. Furthermore, by getting rid of Y from each coordinate, we can obtain a POF expression indicated by the following.

$$y = \frac{1}{\sin\theta(L_x - H'\cos\theta)}\left\{\left(L_x\cos\theta - f - H'\cos^2\theta\right)x - L_x\left(L_x\cos\theta - L_y\sin\theta + H - H'\right)\right. \\ \left. + f\cos\theta(H + H') - H'\sin\theta\left(L_x\sin\theta + L_y\cos\theta\right) + HH'\cos\theta\right\} \tag{1}$$

But, if the distance between an image sensor and a target object is so far, we assume that the lens is thin. For this, the following expression is satisfied.

$$H = H' = 0 \tag{2}$$

Therefore, substituting the above in (1), we obtain the following expression.

$$y = \frac{1}{\sin\theta}\left(\cos\theta - \frac{f}{L_x}\right)x - \frac{L_x\cos\theta - L_y\sin\theta}{\sin\theta} \tag{3}$$

In this case, an image sensor plane, a lens principal plane, and a POF intersect at the following point on the y-axis.

$$\left(0, -\frac{L_x\cos\theta - L_y\sin\theta}{\sin\theta}\right) \tag{4}$$

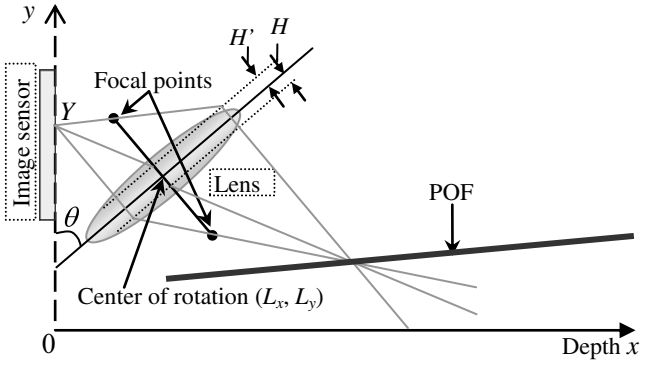


Fig. 2. POF position using a tilted lens

This law is known as the Scheimpflug principle. This condition shows that the POF is inclined to the horizontal by an angle θ of the lens; but our method uses horizontal POF.

There is a way to set a tilted POF by using not only a tilted lens, but also a tilted image sensor. However, the image by the latter setting has distortion. Hence, if final output of an application is a depth map image, the former setting takes advantage.

3.2 DOF by a Tilted POF

By using a normal optical unit, both the plane of image sensor and the plane of lens principal are parallel, and the near DOF limit and far DOF limit are also parallel with respect to these planes. However, in the situation that we use a tilted POF, it looks like different as shown in Fig.3.

Both DOF limits dashed line in Fig.3 with tilted lens are liner as same as with normal optics. If φ is the angle of POF with respect to the ground, the gradient of POF is

$$\tan\varphi = \frac{L_x \cos\theta - f - H'\cos^2\theta}{\sin\theta\left(L_x - H'\cos\theta\right)} \tag{5}$$

Strictly, because a shape of circle of confusion is oval and its size changes by the distance between the lens and the POF, the depth of focus is variable. However, because its variant is slight, we approximate the depth of focus with constant variant g. Hence, both the gradient of near DOF limit and far DOF limit are derived like the following.

$$\begin{aligned}\tan\varphi_+ &= \tan\varphi + \frac{g}{f}\left(\frac{1}{\tan\theta} + \tan\varphi\right)\\ \tan\varphi_- &= \tan\varphi - \frac{g}{f}\left(\frac{1}{\tan\theta} + \tan\varphi\right)\end{aligned} \tag{6}$$

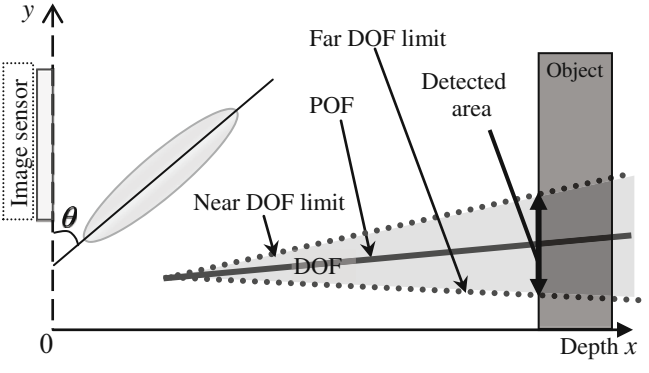


Fig. 3. A titled POF and DOF

Given that the DOF limit has the gradient like (6), and the DOF limit are

$$\begin{aligned} y &= \tan\varphi_+ \cdot x - \frac{f}{\tan\theta} - (L_x + H\cos\theta)\tan\varphi_+ + (L_y - H\sin\theta) \\ y &= \tan\varphi_- \cdot x - \frac{f}{\tan\theta} - (L_x + H\cos\theta)\tan\varphi_- + (L_y - H\sin\theta) \end{aligned} \tag{7}$$

Furthermore, if a thin lens is assumed to be used, it is possible to contribute (2) into (7).

Therefore, we can obtain the DOF size in vertical at a depth (x) by the difference of two equations in (7).

3.3 Depth Estimation by a Horizontal POF

In this section, we describe the proposed method by using the DOF size in vertical that is related with the depth value.

First, a POF is set parallel with respect to the ground ($\varphi = 0$). Thus,

$$\tan\varphi = 0 \quad . \tag{8}$$

Furthermore, the both gradient of near DOF limit and far are, respectively, represented as follows from (6):

$$\begin{aligned} \tan\varphi_+ &= \frac{g}{f\tan\theta} \\ \tan\varphi_- &= -\frac{g}{f\tan\theta} \end{aligned} \tag{9}$$

Contributing (9) into (7), the DOF limit expressions are very simple.

Fig.4 is a relation between depth and DOF size that is calculated from DOF limit difference of (7) with (9); the DOF size is value when it is projected into an image sensor. In real, an optical unit used has a field of view. Hence, as shown in Fig. 4, the projected pixel size of DOF limit difference does not increase in liner according to depth x. It means that the estimated accuracy descends at large depth value. And, a DOF starts from first principal point. Hence, it is not good for estimation accuracy when detectable depth is far and target object is low height. Therefore, we adopt another POF and estimate the overlap area of two DOF limit range (represented by gray zone in Fig. 5). Herewith, we can shift the estimation area to far side, according to an application. Fig. 6 shows an example that we set estimation area from about 600 mm.

In the rest of this section, we describe the process to obtain a depth value from these images. First, we obtain two images by using two POF. For this, we use the following E value, which is based on the derivation of a Laplacian.

$$E = \left| l_{F(x+1,y)} + l_{F(x-1,y)} - 2l_{F(x,y)} \right| + \left| l_{F(x,y+1)} + l_{F(x,y-1)} - 2l_{F(x,y)} \right| \tag{10}$$

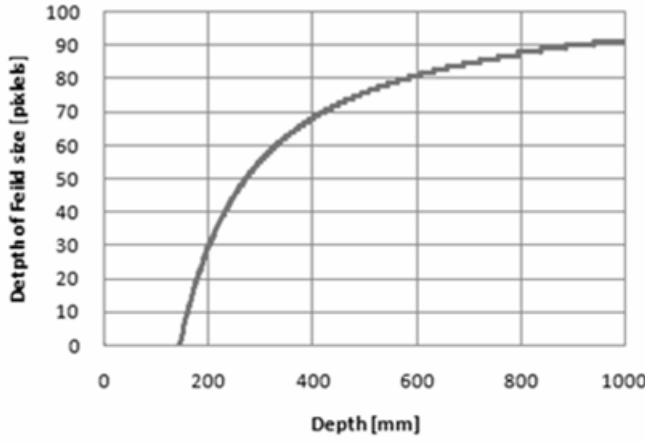


Fig. 4. Relation between depth and DOF size

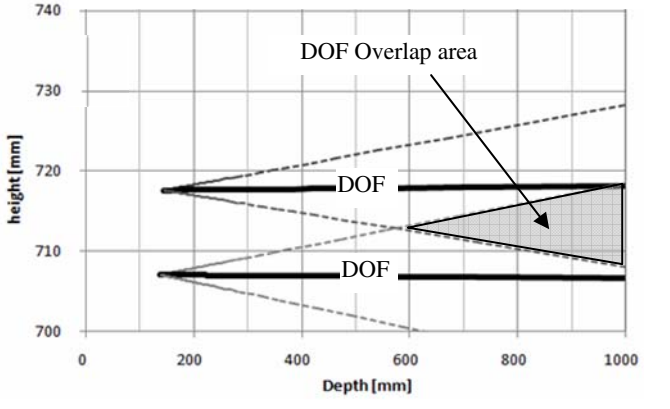


Fig. 5. DOF Overlap area of two POF

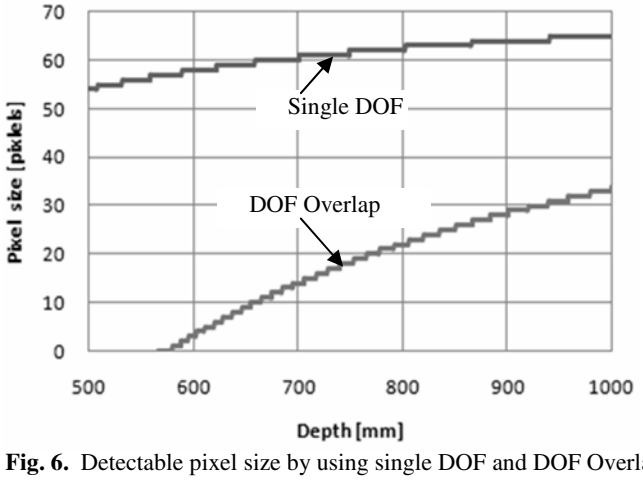


Fig. 6. Detectable pixel size by using single DOF and DOF Overlap

where l is the pixel intensity, F is the focal position, and (x, y) are the pixel coordinates. When E is high value, the focal position is judged to be the in-focus position. This evaluation is sufficiently simple to be implemented on hardware. Thus, the speed can be very high. Then, we can obtain overlap area of DOF by using the AND operation with two in-focus images, and count distribution of in-focus points in vertical. Last, we can get a depth value by checking a table that is predetermined from a relation between depth and in-focus points in advance.

4 Experiment on Our Proposed Method

Fig. 7 shows the camera system used in the experiment and Table I lists the primary specification of the devices used in this system. And, we set two horizontal POFs with the lens that are tilted at angles θ=38.0°, 42.5°, and L_x=118.5 mm,126.2 mm. The distance between these POFs is 10.3 mm. As we treat the optical unit as a thick lens, H=23.9 mm, H'=5.5 mm is set. We used a checked pattern board (a square of 2 mm on a side) as the target object that is planar and placed parallel with respect to the image sensor.

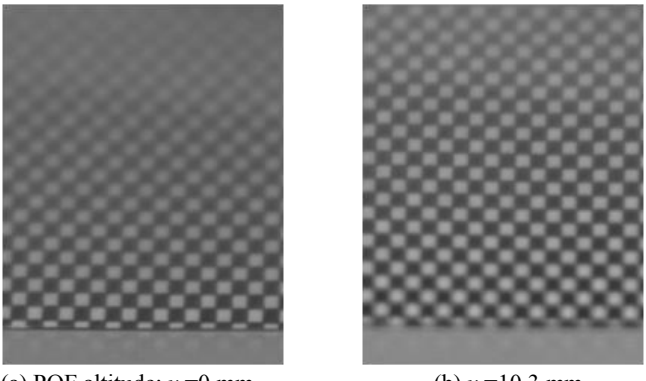
Fig. 7. Camera system used in our experiment

Table 1. Primary specification of items used

Item	*Maker & Model of product*	*Specifications*
Camera	Canon EOS Kiss X2	
	- Image sensor size	22.2 × 14.8 mm
	- Effective pixels	approx. 1220M pixels
	- Total pixels	approx. 1240M pixels
Optical bench	HORSEMAN LD	
	- Minimum flange back	approx. 70 mm
	- Monorail length	400 mm
	- Raise, fall, shift length	30 mm
	- Swing, tilte angle	360°
Lens module	FUJIFILM FUJINON SWD	
	- Focal length	90 mm
	- Apperture ratio	5.6
	- Minimum diaphragm	64

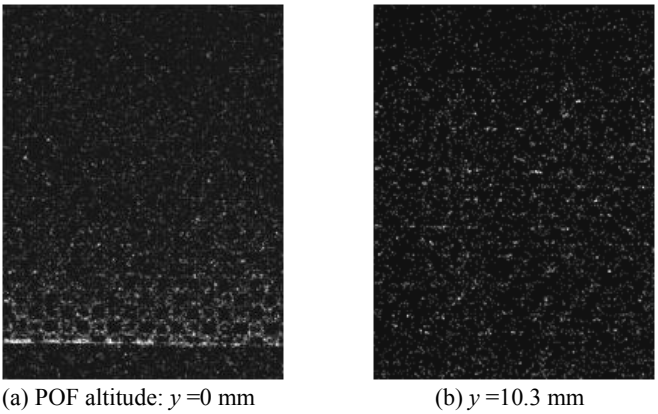
The distance between the camera and object is 750 mm. At each frame, the object goes ahead 50mm toward far side from the camera. Hence we captured eight images that are taken at four depth positions (750 mm, 800 mm, 850 mm, 900 mm) by two POFs individually. Fig. 8 shows the captured images using two horizontal POFs: One has attitude 0.0 mm, while the other is 10.3 mm. In this figure, a variation in the blur appears on each surface of the object.

Then, the edge-detected images (Fig. 9) are generated from the individual images like Fig. 8 by the calculation using (10). It means that we can find the DOF region by identifying the in-focus points and out-of-focus regions. Next, the images of DOF overlap are generated by the AND operation with each edge-detected image pairs (Fig.9) at the every depth.



(a) POF altitude: y =0 mm (b) y =10.3 mm

Fig. 8. The part of magnified image by horizontal POF (depth: x=750 mm)



(a) POF altitude: y =0 mm (b) y =10.3 mm

Fig. 9. The part of magnified edge-detected image which represent in-focus points of Fig.8 (depth: x=750 mm)

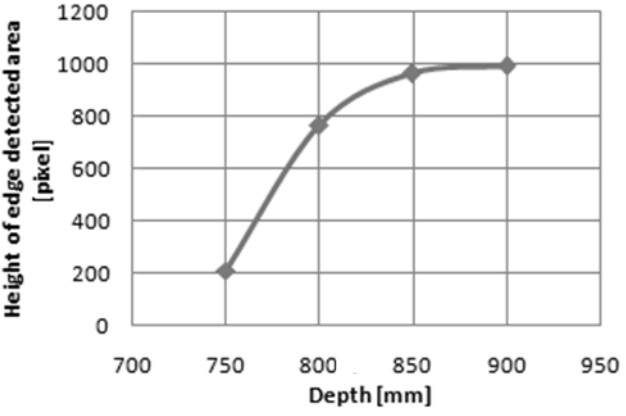


Fig. 10. Depth and height of detected DOF overlap

Fig. 10 shows a relation between the depth and the DOF overlap value from our experiment. By a comparison of Fig.10 with the theoretical value shown in Fig. 6, the curb of line indicates the same aspect. However, detectable pixel value of our experiment is different from theoretical value. It is possible for the theoretical graph to change greatly, because it depends on setting the diameter of circle of confusion: g in (6), and the threshold value when we generated binarized images. For that reason, when we use our method in real application, we need to prepare a data table that is relation between the depth and the DOF overlap value in the real environment in advance.

In additional, if the target objects are almost planar and placed parallel to the image sensor, it is also possible to generate a rough depth map; especially under the condition in which targets are limited like in-vehicle application. Because, for regions that do not possess a depth value, the depth is interpolated using the depth of the in-focus pixels. We have already reported our interpolation method in [7] that involves the further details.

As mentioned, we obtained depth by only using overlap value of DOF. Needless to add, we can also use each POF value with the overlap value to extend detectable range.

5 Conclusion

In this paper, we proposed the depth estimation method using variant of DOF with the horizontal POFs. We succeeded in reducing largely the number of multiple focus images with enough simple calculations. Herewith, we can get rid of the drawback of conventional passive method for real-time usage. Additionally, we confirmed the availability of our proposed method by a simple experiment.

We are planning to improve the DOF detection algorithm using inter-frame difference. Additionally, we are going to construct an application using our method.

References

1. Oike, Y., Ikeda, M., Asada, K.: A 375 365 High-Speed 3-D Range-Finding Image Sensor Using Row-Parallel Search Architecture and Multisampling Technique. IEEE Journal of Solid-State Circuits 40(2), 444–453 (2005)
2. Kawahito, S., Halin, I.A., Ushinaga, T., Sawada, T., Homma, M., Maeda, Y.: A CMOS Time-of-Flight Range Image Sensor With Gates-on-Field-Oxide Structure. IEEE Sensors Journal 7(12), 1578–1586 (2007)
3. Krotkov, E.: Focusing. International Journal of Computer Vision 1(3), 223–237 (1987)
4. Nayar, S.K., Nakagawa, Y.: Shape from Focus. IEEE Transactions on Pattern Analysis and Machine Intelligence 16(8), 824–831 (1994)
5. Subbarao, M., Surya, G.: Depth from Defocus: A spatial domain approach. International Journal of Computer Vision 13(3), 271–294 (1994)
6. Kashiyama, H., Yoshida, T., Hamamoto, T.: Smart Image Sensor for Detection of Well-focused position using Multiple Focused Images. In: ISPACS, vol. D2-6, pp. 282–285 (2003)
7. Ikeoka, H., Kashiyama, H., Hamamoto, T., Kodama, K.: Depth Estimation by Smart Imager Sensor using Multiple Focus Images. The Institute of Image Information and Television Engineers 62(3), 384–391 (2008)
8. Ikoeka, H., Hamamoto, T.: High Speed Depth Estimation Using Tilted Focal Planes. In: 4th IEEE International Symposium on Electronic Design, Test and Applications (DELTA), pp. 175–178 (2008)
9. Beucher, S.: Geodesic Reconstruction, Saddle Zones & Hierarchical Segmentation. Image Anal Stereol 20, 137–141 (2001)

A Study on Stereo and Motion Data Accuracy for a Moving Platform

Sandino Morales[1], Young Woon Woo[2], Reinhard Klette[1], and Tobi Vaudrey[1]

[1] University of Auckland, Auckland, New Zealand
pmor085@aucklanduni.ac.nz
http://www.mi.auckland.ac.nz/
[2] Dong-Eui University, Department of Multimedia Engineering, Busan, Korea

Abstract. Stereo and motion analysis are potential techniques for providing information for control or assistance systems in various robotics or driver assistance applications. This paper evaluates the performance of several stereo and motion algorithms over a long synthetic sequence (100 stereo pairs). Such an evaluation of low-level computer vision algorithms is necessary, as moving platforms are being used for image analysis in a wide area of applications. In this paper algorithms are evaluated with respect to robustness by modifying the test sequence with various types of realistic noise. The novelty of this paper is comparing top performing algorithms on a long sequence of images, taken from a moving platform.

1 Introduction

The main task of computer vision is to use image data recorded by one or multiple cameras to understand the given 3D environment. In particular, stereo algorithms obtain 3D information about the scene geometry, and motion algorithms gather information about the 2D motion of the images. Both types of information are needed to reconstruct the 3D motion of the scene. These algorithms still represent a challenging task for the vision community. For mobile devices the challenge becomes even more difficult; moving background, change in lighting conditions, possible misalignments of cameras, and so forth, make the task of the algorithms even harder. However, the use of cameras has become a popular data sensor for moving vehicles. Vision-based stereo has already been used in a wide variety of vehicles, including wheelchairs (e.g., for the detection of obstacles, unevenness of the ground, detection of stairs and ropes or beams in the air [13]), or for forklifts, where the operator gets valuable information to deal with heavy loads at great heights [14], and in standard cars to assist a driver while driving on a road.

Thus, it is necessary to evaluate the performance of these algorithms to detect which one performs the best in different situations and to encourage their theoretical improvement. Several authors have evaluated stereo algorithms; for example [1] presented one of the earliest evaluations of stereo algorithms and [3] presented one of the most recent and representative evaluation papers so far. Several stereo algorithms were tested, but in both cases the experiments were

J.-H. Kim et al. (Eds.): FIRA 2009, LNCS 5744, pp. 292–300, 2009.

done with small sets of images. Motion algorithms have also been evaluated, the approach presented in [17] influenced the evaluation of motion algorithms until recently. Now, [16] is the main approach for testing and comparing algorithms online. However, both publications focus on very short image sequences.

In this paper we evaluate the performance of stereo and motion algorithms over a long (i.e., 100 stereo pair) sequence. The analysis of long sequences allows the usage of temporal information (e.g., [5]). In order to test the robustness of the chosen algorithms, we added different kinds of noise to the sequence so that algorithms can be tested under different conditions [12]. As we are interested in testing the algorithms on a mobile platform (a wheelchair or car), we used a long sequence that is publicly available, with ground truth for motion and stereo, in Set 2 on the *.enpeda..* Image Sequence Analysis Test Site [4]. This sequence simulates a driving situation.

This paper follows [12] with respect to stereo algorithm evaluation, and extends these studies by including evaluations of optic flow algorithms on the same synthetic sequence.

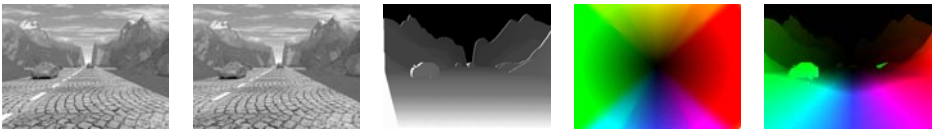
2 Stereo and Optic Flow Algorithms

Stereo Vision: is the process of understanding the 3D information of the environment from the available 2D data (images) by matching the corresponding projections of a 3D point in (at least) two images. The algorithms chosen for our analysis are as follows:

(i) *Dynamic programming stereo*: we compare a standard algorithm [6] (DP), against one with temporal (DPt), spatial (DPs), or temporal and spatial (DPts) propagation; see [7] for propagation details.
(ii) *Belief propagation stereo* (BP): we use a coarse-to-fine algorithm of [8] with quadratic cost function, as reported in [9];
(iii) *Semi-global matching* (SGM): characterizes one of the top performing stereo strategies, see [2]; we chose two cost functions, mutual information (SGM MI) or Birchfield-Tomasi (SGM BT) [10];

Motion Analysis: is estimated from a pair of images taken at slightly different times. Optical flow algorithms aim to detect the visible displacement of pixels in the image plane to understand the motion of the 2D projection of 3D motion of the visible objects (and background). The following are the algorithms used in our evaluation:

(i) *Horn-Schunck algorithm* (HS): we use the program as available in the OpenCV library [11];
(ii) *Combination of Local and Global* (CLG): optimization as in [18]; we use an implementation from the *.enpeda..* group; see acknowledgment;
(iii) *BBPW*: this is named after the initials of surnames of all the four co-authors of [15], we use an implementation also from the *.enpeda..* group; see acknowledgment.



(a) Left image (b) Right image (c) GT Disparity (d) Color Key (e) GT Flow

Fig. 1. Stereo image pair #40 of the sequence; (a) and (b) are original left and right images. (c) ground truth data in gray-scale encoding: light = close, dark = far, white = occlusion. (d) color key for encoding optic flow. (e) ground truth optic flow.

Evaluation Approach: The algorithms were tested using the original sequence and with the same sequence corrupted with different types and magnitudes of noise. For the stereo algorithms we analyzed the stereo pair at each time frame, and for the motion algorithms only the left images.

Data Set and Visualization: For our experiments we use a long sequence of 100 synthetic stereo image pairs and ground truth data (for stereo and motion algorithms), which are all available on [4]. To visualize the stereo results we use gray scale encoding: light for closer objects and dark for objects further away. The color key that we use to visualize the motion results uses hue for direction and intensity for vector size; dark to light means small to large as seen in Figure 1(d).

Noise: A mobile platform has to deal with non-controlled environments. Thus, we consider it necessary to test the robustness of the algorithms in different situations. Therefore, we corrupt our data set with three different kinds of noise: brightness differences, blurring, and Gaussian white noise. As a consequence of the movement of the platform, brightness on images can change from one frame to another or even between the left and right image in the same frame of a stereo sequence. Blurring may be caused by differences in the focus of the lenses due to movements of the platform. Gaussian noise is present in images taken, even with modern camera technology. Note that we are aware that this may not be an extensive noise list, but it is sufficient to show the importance of testing algorithms in different conditions.

To alter the brightness of the images, we add a constant brightness c to each pixel of every image. Blurring was applied to the sequence by convolving the images with a Gaussian smoothing kernel of size k. Finally, the Gaussian noise was generated by adding at each pixel random Gaussian (normal distribution) white-noise $\mathcal{N}(\mu, \sigma)$, with a mean μ of zero, and a varying standard deviation σ. The parameters are varied over the sequence and presented in Table 1.

Table 1. Parameters of the different kind of noise added to the sequence

	Left Image		Right Image	
Noise Method	$1 \leq t \leq 50$	$51 \leq t \leq 100$	$1 \leq t \leq 50$	$51 \leq t \leq 100$
Brightness	$c = t - 50$		$c = 50 - t$	
Gaussian Noise	$\sigma = t$	No noise	$\sigma = t$	$\sigma = 101 - t$
Gaussian Blur	$k = 2\,t - 1$	No noise	$k = 2\,t - 1$	$k = 203 - 2\,t$

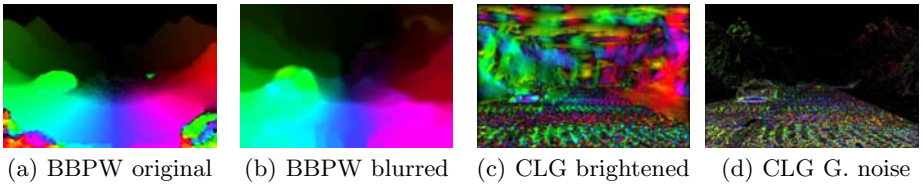
To evaluate the motion algorithms we modify the left images using the parameters defined for the right images (Table 1), with the exception of the brightness constant: where $c = t - 52$ is used for even t and $c = 51 - t$ for odd t.

Quality Metrics: Following [12] and [3], for the evaluation of stereo algorithms we use Root Mean Squared (RMS) and % Bad Percentage Pixels (BPP). For BPP we used two different thresholds. Motion algorithms were evaluated using the Average Angular Error (AAE) and the End Point Error (EPE) as quality metrics, also used (for example) in [16]. The metrics used here allow analysis performance similar to the approach at Middlebury [2], but as we are working with long sequences, we can make statistical inference from the obtained data, such as the mean, zero-mean variance, maximum and minimum for each error metric over the sequence.

3 Results

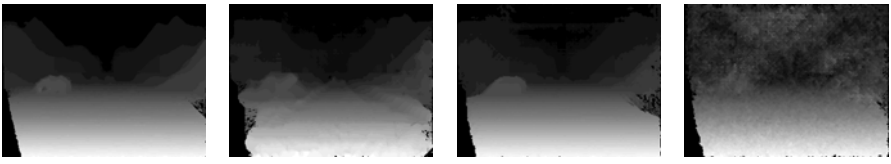
As this is a brief conference publication, only the results for AAE for motion algorithms are presented. For stereo algorithms, just the RMS results for the original and bright altered sequences are shown; however, for our discussion we use the results obtained from all four of the metrics. We include resultant images obtained with the algorithms with all kinds of noise (see Figures 2 and 3). For a more detailed report concerning the stereo algorithms, see [12] (and more detailed results of the motion algorithms are available on the Technical Report No. 32 from [4]).

Noise-Free Results: The results obtained with the original sequence are the base for the robustness analysis, see Figures 4(a) and 6(a). For motion algorithms BBPW perform the best with both metrics followed by CLG. In the AAE graph it can be observed a considerable large difference in the magnitude among BBPW and the other two algorithms. It is worth to say that for EPE, CLG performs better than BBPW, and its range is the minimum one of all three techniques. For stereo algorithms, the best algorithms for both metrics are the SGM ones, followed by BP and finally the DP algorithms. The difference in magnitudes between the SGM algorithms and the other ones is noticeable. The best algorithm was SGM BT and the best among the DP algorithms is DP closely followed by the other three. For these sequences RMS and BPP show almost the same information.



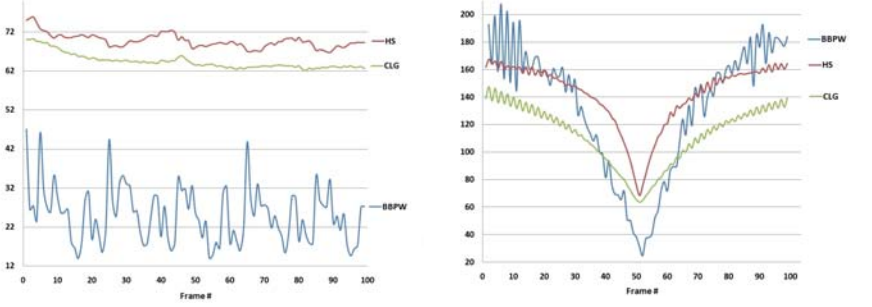
(a) BBPW original (b) BBPW blurred (c) CLG brightened (d) CLG G. noise

Fig. 2. Results of top motion algorithms, with respect to EPE mean, for the original sequence and every kind of noise, on left images 41-42



(a) SGM-BT: original sequence. (b) SGM-BT: blurred sequence. (c) SGM-MI: brightness altered. (d) SGM-BT: Gaussian noise added

Fig. 3. Stereo results of the top performing algorithms, for various kinds of noise and the original image sequence, here shown for image pair #40



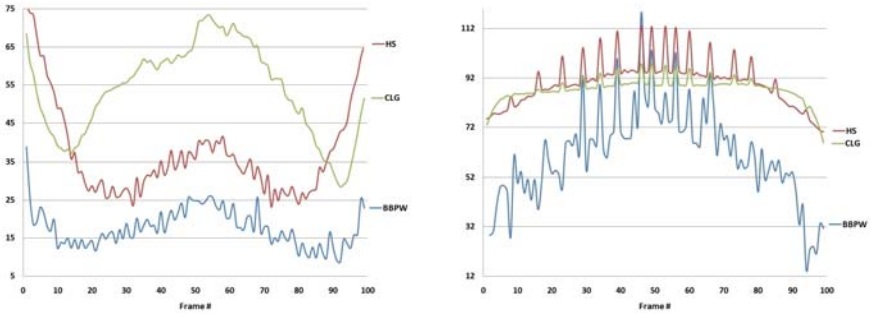
(a) AAE on original sequence. (b) AAE on brightness altered sequence.

Fig. 4. AAE results on original and brightness altered sequence

Gauss Blur Results: BBPW outperforms the other motion algorithms with both metrics. With AAE all the algorithms improve their performance with respect to the noise-free results except for CLG when the blurring is maximum, see Figures 5(a) and 4(a); the change in magnitude is highly noticeable. The improvement is most likely because all the algorithms improve their performance on the road area. The three algorithms behave the best when the amount of blur is medium and peak when the blur is minimum and maximum. The same behavior is observed with EPE.

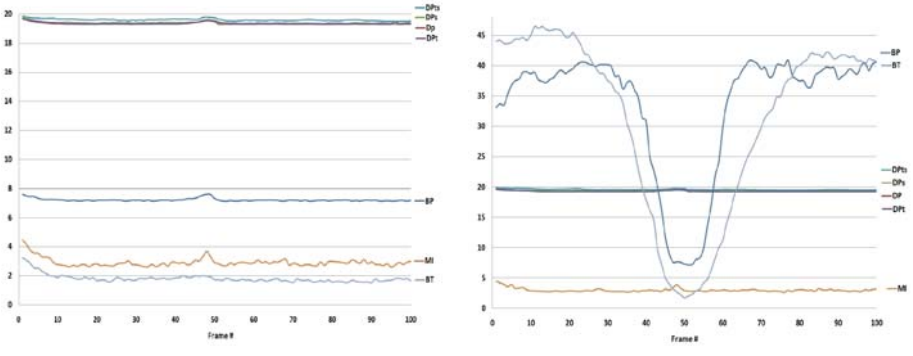
The interesting observation for the stereo algorithms is that when the blurring is in both images, the results are not so bad, but when the blurring is removed from the left image, the results get worse for all algorithms. Again SGM BT is the best, SGM MI seems to have the same problem if both images are blurred, or just one. For BPP the interesting point is that both DP and DPt ranked higher than SGM MI.

Brightness Difference Results: This was the noise that had the biggest impact on the results for both stereo and motion algorithms. The ones that perform the best with the original sequence are the worst in this case (for all the respective metrics). SGM BT and BP are tremendously affected, while SGM MI and the dynamic programming algorithms are relatively robust to this kind of



(a) AAE on blurred sequence. (b) AAE on Gauss. noise added sequence.

Fig. 5. Motion algorithm results for blurred and Gaussian noise sequence



(a) RMS on original sequence. (b) RMS on brightnress altered sequence

Fig. 6. RMS stereo results on the original and blurred sequences

noise. For the flow algorithms, CLG was the best and BBPW produced useless data until the difference in brightness is around 10%. See Figures 4(b) and 6(b).

Gauss Noise Results: The algorithms are very sensitive to this kind of noise too. In this case BBPW was the best for AAE and CLG for EPE, see Figure 5(b). For the latter metric there is a noticeable overlapping in the graphs for all of the algorithms. For stereo, SGM BT is the best algorithm, and among dynamic programming algorithms, DPt is the best and DP the worst. The difference between them is enough to make DPt the best overall (see Table 2(b)) dynamic programming algorithm. SGM BT and BP are relatively robust to this alteration of the images.

Summary: In the Table 2(a) we present the overall statistics for the motion algorithms when the results are analyzed with AAE. BBPW is clearly the best (respect to the mean and standard deviation) for the sequences analyzed here. However, note that its dynamic range is the largest one, due to the very bad

Table 2. Overall results of the four sequences

(a) Overall motion results for AAE.

Algorithm	Mean	St. Dev.	Min.	Max.
BBPW	59.17	158.01	8.65	207.73
CLG	79.57	167.52	28.45	147.59
HS	84.72	187.83	23.16	167.36

(b) Overall stereo results for RMS.

Algorithm	Mean	St. Dev.	Min.	Max.
SGM MI	7.23	20.45	2.61	23.95
SGM BT	11.21	36.51	1.52	46.45
BP	15.41	39.73	6.99	40.92
DPt	20.56	41.47	19.30	29.49
DP	21.00	42.37	19.30	30.28
DPs	21.40	43.37	19.32	32.63
DPts	21.54	43.58	19.48	32.69

results that were obtained with the brightness altered sequence. The bad results with this sequence were compensated with the improvement on the blurred sequence. It is worth noting that for EPE, in the overall statistics (not shown in this report), BBPW was the worst algorithm, once again due to results obtained with the bright altered sequence. CLG was the best for this metric. Finally, HS show an improvement with the blurred sequence, but it has also a bad performance with the brightness altered sequence.

For stereo algorithms (see Table 2(b) for RMS results) the one with better overall performance was SGM MI, with all the metrics used (for stereo algorithms) in this paper. Note that SGM BT outperform best than SGM MI in all the sequences except in the brightness altered one, where its performance was not good at all. The DP algorithms where the worst ones with RMS, with DPt performing the best among them. It is worth to say that DPt has a better overall performance than SGM MI for the Bad Pixel metrics. BP was always below the two best algorithms, showing its worst performance in the bright sequence as with SGM BT.

We briefly report about the average computational time of all the algorithms evaluated in this study. The algorithms were processed on an Intel Core 2 duo E800 at 3.00Ghz under a Windows platform; except for the SGM algorithms (the only time-optimized ones) which were processed on an Intel Centrino Duo at 2.4Ghz using a Linux OS. The dynamic programming algorithms process, on average, one stereo pair in 63.615 s, with DP being the fastest and DPts the slowest. BP was the slowest of the stereo algorithms with a processing time of 106.2235 s. SGM MI processed a stereo pair in 3.974 s. The fastest of the stereo algorithms was SGM BT with 0.741s per pair.

For the motion algorithms, CLG was the fastest processing one pair of images in 27.780 s; this is followed by HS, whose computing time was 39.038 s. The slowest motion algorithm was BBPW, with 90.383 s per pair of images.

4 Conclusions and Future Work

In this paper we presented an approach to evaluate the robustness of stereo and motion algorithms over a long synthetic sequence. In order to do this we tested

several algorithms over a long synthetic sequence, which was corrupted with different kinds of noise. From our results it is clear that most of the algorithms are very sensitive to brightness differences. This has to be highlighted as changes in illumination is one of the most common problems that mobile devices have to deal with. The SGM BT stereo algorithm, whose results were the worst with this type of noise, was the best in all the other sequences. A similar behavior presented the BBPW motion algorithm. As a direct consequence of using long sequences we were able to observe that DPt represent a good option for the dynamic programming algorithms. The future work will include a wider set of noise types, more challenging sequences (real ones), a more in depth study on the quality metrics and a way to evaluate precisely the performance of the algorithms when there is no ground truth available.

Acknowledgments. The authors acknowledge valuable support by members of the *.enpeda..* team at Auckland university, in particular by Xi (Todd) Yang, James Milburn, and Xuan (Sean) Guo.

References

1. Mohan, R., Medioni, G., Nevatia, R.: Stereo error detection, correction, and evaluation. In: Pattern Analysis and Machine Intelligence, vol. 11, pp. 113–120 (1989)
2. Mohan, R., Medioni, G., Nevatia, R.: Middlebury stereo and motion evaluation, `http://vision.middlebury.edu/`
3. Scharstein, D., Szeliski, R.: A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. Int. J. of Computer Vision 47, 7–42 (2002)
4. .enpeda.. image sequence analysis test site (EISATS), `http://www.mi.auckland.ac.nz/EISATS/`
5. Vaudrey, T., Badino, H., Gehrig, S.K.: Integrating disparity images by incorporating disparity rate. In: Sommer, G., Klette, R. (eds.) RobVis 2008. LNCS, vol. 4931, pp. 29–42. Springer, Heidelberg (2008)
6. Ohta, Y., Kanade, T.: Stereo by two-level dynamic programming. In: Proc. IJCAI, pp. 1120–1126 (1985)
7. Liu, Z., Klette, R.: Dynamic programming stereo on real-world sequences. In: Proc. ICONIP. LNCS (2008)
8. Felzenszwalb, P.F., Huttenlocher, D.P.: Efficient belief propagation for early vision. Int. J. Computer Vision 60, 261–268 (2006)
9. Guan, S., Klette, R., Woo, Y.W.: Belief propagation for stereo analysis of night-vision sequences. In: Wada, T., Huang, F., Lin, S. (eds.) PSIVT 2009. LNCS, vol. 5414, pp. 932–943. Springer, Heidelberg (2009)
10. Hirschmüller, H.: Accurate and efficient stereo processing by semi-global matching and mutual information. In: Proc. Computer Vision and Pattern Recognition (CVPR), vol. 2, pp. 807–814 (2005)
11. Open Computer Vision Library (OpenCV), `http://sourceforge.net/projects/opencvlibrary/`
12. Morales, S., Vaudrey, T., Klette, R.: An in depth robustness evaluation of stereo algorithms on long stereo sequences. Tech. Rep. TR-30, University of Auckland (2008), `http://mi.auckland.ac.nz`

13. Satoh, Y., Skaue, K.: An Omnidirectional Stereo Vision-Based Smart Wheelchair. EURASIP Journal on Image and Video Processing 2007, 11, Article ID 87646
14. Seelinger, M., Yoder, J.D.: Automatic Pallet Engagement by a Vision Guided Forklift. In: Proc. International Conference on Robotics and Automation (ICRA), pp. 4068–4073 (2005)
15. Brox, T., Bruhn, A., Papenberg, N., Weickert, J.: High accuracy optical flow estimation based on a theory for warping. In: Pajdla, T., Matas, J(G.) (eds.) ECCV 2004. LNCS, vol. 3024, pp. 25–36. Springer, Heidelberg (2004)
16. Baker, S., Scharstein, S., Lewis, J.P., Roth, S., Black, M.J., Szelisky, R.: A Database and Evaluation Methodology for Optical Flow. In: Proc. IEEE 11th International Conference on Computer Vision (ICCV), pp. 1–8 (2007)
17. Barron, J., Fleet, D., Beauchemin, S.: Performance of Optical Flow Techniques. International Journal of Computer VIsion (IJCV) 12(1), 43–77 (1994)
18. Bruhn, A., Weickert, J., Schnrr, C.: Lucas/Kanade meets Horn/Schunck: Combining local and global optic flow methods. Updated version with errata. Int. J. Computer Vision 61, 211–231 (2005)

Equivalent Relationship of Feedforward Neural Networks and Real-Time Face Detection System

Shuzhi Sam Ge, Yaozhang Pan, Qun Zhang, and Lei Chen

Social Robotics Lab, Interactive and Digital Media Institute, Department of Electrical and Computer Engineering, National University of Singapore, 117576
Tel.: (65) 6516-6821; Fax. (65) 6779-1103
elegesz@nus.edu.sg
http://robotics.nus.edu.sg/sge/

Abstract. In this paper, we mainly investigate a fast algorithm, Extreme Learning Machine (ELM), on its equivalent relationship, approximation capability and real-time face detection application. Firstly, an equivalent relationship is presented for neural networks without orthonormalization (ELM) and orthonormal neural networks. Secondly, based on the equivalent relationship and the universal approximation of orthonormal neural networks, we successfully prove that neural networks with ELM have the property of universal approximation, and adjustable parameters of hidden neurons and orthonormal transformation are not necessary. Finally, based on the fast learning characteristic of ELM, we successfully combine ELM with AdaBoost algorithm of Viola-Jones in face detection applications such that the whole system not only retains a real-time learning speed, but also possesses high face detection accuracy.

1 Introduction

Face detection is one of the most popular research topics in pattern recognition during last decade [1, 2], which is directly relevant to the face recognition and human-robot interaction. It can be widely used in entertainment, information security, intelligent robotics and so on. Face detection system can be defined as follows: given any image input, which could be a digitized video signal or a scanned photograph, determines whether or not there are human faces in the image, and if there are, returns their locations. Since faces can appear potentially anywhere in the images, a well designed face detection system normally requires a high frame rate, a low false detection rate, a wide range of working environment, and the ability to cope with different scales, locations, orientations, occlusions and lighting conditions. Among all the face detection methods, one of the most impressive studies is AdaBoost algorithm [3,4], which uses the concept of an "integral image", along with a rectangular feature representation and an AdaBoost algorithm as its learning method, to detect faces at 15 frames per second. This represents an improvement in the computation time of an order of magnitude over previous implementations of face detection algorithms.

However, one disadvantage of the AdaBoost algorithm is that it cannot distinguish face-like images from the real face images because those face-like images will pass every stage. The reason comes from lack of non-face image database. This can result in

J.-H. Kim et al. (Eds.): FIRA 2009, LNCS 5744, pp. 301–310, 2009.

a brittle classifier since a non-face image can be misclassified because it totally satisfy all the criteria of each single stage. As we know, imprecise human face detection will lead to the insufficient interactions between humans and social robots, even for false interpretations or conclusions.

Based on this observation, we consider applying neural networks to face detection due to their approximation capability. Since face detection can be treated as a two class pattern recognition problem, various neural network architectures have been proposed to improve face detection performance. Especially, single-hidden-layer feedforward neural networks (SLFNs) have been successfully applied to a wide range of real applications due to their approximation capability [5, 6], where many researchers have shown that SLFNs can approximate target functions with a finite number of neurons.

Recently, an effective and non-iterative algorithm, Extreme Learning Machine (ELM), has been proposed in [7]. The basic principle of ELM is randomly generating hidden neurons (the input-to-hidden parameters are generated based on a continuous distribution) and analytically determining the hidden-to-output weights of neural networks one time by using Moore-Penrose generalized inverse. In fact, according to the matrix theory, the hidden-to-output weights determined by the inverse of hidden matrix is a least-square solution for the whole neural network. Due to the random character of hidden neurons, ELM breaks through the drawback of conventional neural network algorithms, and has been proven to be an effective method to make neural networks achieve a better generalization performance at a fast learning speed. Thus we naturally combine real-time ELM algorithm with Viola and Jones's work to exclude those false face images, which not only improves the face detection accuracy, but also retains the real-time learning speed at the same time.

Though many experimental results show that the input-to-hidden neurons do not need to be adjusted at all, so far there lacks a rigorous theoretical justification whether SLFNs with ELM are universal approximator. In this paper, we systematically analyze ELM to demonstrate the equivalent relationship between ELM and orthonormal neural networks, and prove the universal approximation of ELM.

The main contributions of this paper are summarized as follows:

i) The presentation of an equivalent relationship between neural networks without orthonormalization (ELM) and orthonormal neural networks.
ii) The proof of the universal approximation of orthonormal neural networks.
iii) The deduction of the universal approximation of ELM based on the equivalent relationship.
iv) The combination of ELM and AdaBoost for face detection system, which not only retains a real-time learning speed, but also has high face detection accuracy.

2 Preliminaries

Before we present our main results, we need to introduce some terminologies and background information.

2.1 Approximation with Orthonormal Basis

Let $L^2(X)$ be a space of functions f on a compact subset X in the n-dimensional Euclidean space $\mathbf{R}^n$ such that $|f|^2$ are integrable, i.e., $\int_X |f(\mathbf{x})|^2 d\mathbf{x} < \infty$. Assuming that the training samples are generated with a uniform probability distribution and all the functions belong to the space $L^2(X)$. Similar to [8], for $u, v \in L^2(X)$, the inner product of $u(\mathbf{x})$ and $v(\mathbf{x})$ is defined as $\int_{uv} = \int_X u(\mathbf{x})v(\mathbf{x})d\mathbf{x}$. The norm in $L^2(X)$ space is denoted as $\|\cdot\|$. Based on [9] (p. 1179), the inner product of two functions can be approximated by

$$\widehat{\int}_{uv} = \sum_{i=1}^{N} u(\mathbf{x}_i)v(\mathbf{x}_i) \tag{1}$$

where N is the number of training samples. We name it as approx-inner product. From mathematical viewpoint, equation (1) means that for any $\varepsilon > 0$, there exists a $N_0 > 0$ such that when the number of training samples $N > N_0$, we have

$$|\int_{uv} - \widehat{\int}_{uv}| < \varepsilon \tag{2}$$

The nonzero vectors $\mathbf{e}_1, \cdots, \mathbf{e}_K$ are orthogonal if $\int_{\mathbf{e}_i \mathbf{e}_j} = 0$, for $i \neq j$; $i, j = 1, \cdots, K$ and orthonormal if they are orthogonal and all the vectors have unit length, $\int_{\mathbf{e}_i \mathbf{e}_i} = 1$, for $i = 1, \cdots, K$ [10].

Lemma 21. *[11] For arbitrary* $\mathbf{x} = [x_1, x_2, \cdots, x_n]^T \in C^n$ *and* $\mathbf{A} = [a_{ij}]_{i,j=1}^n \in C^{n \times n}$, *let* $\|\mathbf{x}\|_1 = \sum_{i=1}^n |x_i|$ *and* $\|\mathbf{A}\|_1 = \max_{1 \leq j \leq n}^n \sum_{i=1}^n |a_{ij}|$, *we have*

$$\|\mathbf{A}\mathbf{x}\|_1 \leq \|\mathbf{A}\|_1 \|\mathbf{x}\|_1$$

Lemma 22. *[12] For any* f *and* $\{e_i\}_{i=1}^{\infty}$ *in Hilbert space* $\mathscr{H}$, *the identity*

$$\|f\|^2 = \sum_{j=1}^{\infty} |a_i|^2, \text{ where } a_i = \int_{f \mathbf{e}_i},$$

which is called **Parseval identity**, *holds if and only if* $\{e_i\}_{i=1}^{\infty}$ *is an orthonormal basis.*

Lemma 23. *[12] Any Hilbert space* $\mathscr{H}$ *has an orthonormal basis.*

Lemma 23 ensures that there exists an orthonormal basis at least in Hilbert space. L^2 space is only a special case of Hilbert space, therefore restricting all the functions into L^2 space in this paper is meaningful. The following theorem proves the approximation capability under an orthonormal basis:

Theorem 1. *For any* $f \in \mathscr{H}$, *suppose* $\{e_1, e_2, \cdots\}$ *is an orthonormal basis set in* $\mathscr{H}$, *then*

$$\lim_{n \to \infty} \|f - \sum_{j=1}^{n} \int_{f \mathbf{e}_j} e_j\| \to 0$$

Proof. Set $\bar{f} = f - \sum_{j=1}^{\infty} \int_{f\mathbf{e}_j} e_j$. Since Hilbert space $\mathscr{H}$ is a complete inner product space [10], we have $\bar{f} \in \mathscr{H}$. Since $\{e_1, e_2, \cdots\}$ is an orthonormal basis in $\mathscr{H}$, then according to Parseval identity in Lemma 22, we have

$$\begin{aligned}\|\bar{f}\|^2 &= \sum_{k=1}^{\infty} |\int_{\widehat{f}\mathbf{e}_k}|^2 = \sum_{k=1}^{\infty} |\int_{(f-\sum_{j=1}^{\infty}\int_{f\mathbf{e}_j} e_j)\mathbf{e}_k}|^2 \\ &= \sum_{k=1}^{\infty} |\int_{f\mathbf{e}_k} - \int_{(\sum_{j=1}^{\infty}\int_{f\mathbf{e}_j} e_j)\mathbf{e}_k}|^2 = \sum_{k=1}^{\infty} |\int_{f\mathbf{e}_k} - \int_{f\mathbf{e}_k}|^2 = 0\end{aligned}$$

2.2 Single Hidden Layer Feedforward Neural Networks (SLFNs)

The output of a SLFN with L hidden neurons can be represented by

$$f_L(\mathbf{x}) = \sum_{i=1}^{L} \beta_i g_i(\mathbf{x}) \tag{3}$$

where $g_i(\mathbf{x}) = g(\mathbf{a}_i, b_i, \mathbf{x})$ denotes the i-th hidden neuron output function.

A standard SLFN with L hidden neurons can learn N arbitrary distinct samples $(\mathbf{x}_j, \mathbf{t}_j)$, $j = 1, \cdots, N$, with *zero error* means that there exist parameters $\mathbf{a}_i$, b_i and β_i for $i = 1, \cdots, L$ such that $\sum_{j=1}^{N} \|\mathbf{o}_j - \mathbf{t}_j\| = 0$. Therefore, a desirable objective of designing a SLFN is to find proper parameters such that

$$\sum_{i=1}^{L} \beta_i g_i(\mathbf{x}_j) = \mathbf{t}_j,\ i = 1, \cdots, L,\ j = 1, \cdots, N \tag{4}$$

These N equations can be condensed as

$$\mathbf{H}\beta = \mathbf{T} \tag{5}$$

where $\beta = [\beta_1, \cdots, \beta_L]^T$, $\mathbf{T} = [t_1, \cdots, t_N]^T$ and

$$\mathbf{H} = \begin{bmatrix} g_1(\mathbf{x}_1) & \cdots & g_L(\mathbf{x}_1) \\ \vdots & \ddots & \vdots \\ g_1(\mathbf{x}_N) & \cdots & g_L(\mathbf{x}_N) \end{bmatrix}_{N \times L}$$

where the matrix $\mathbf{H}$ is called as the hidden layer matrix of the SLFN.

As discussed in [7], the parameters of hidden neurons need not be tuned and can be randomly generated permanently according to any continuous probability distribution. Equation (5) then becomes a linear system and the output weights $\hat{\beta}$ are estimated as

$$\hat{\beta} = \mathbf{H}^{\dagger}\mathbf{T} \tag{6}$$

where $\mathbf{H}^{\dagger}$ denotes Moore-Penrose generalized inverse of $\mathbf{H}$ [13,14]. In fact, such hidden-to-output weights determined by Moore-Penrose generalized inverse are actually a least square solution for the whole system $\mathbf{H}\beta = \mathbf{T}$. The fast learning algorithm is called as Extreme Learning Machine (ELM) in [7].

3 Approximation of ELM

In this section, we will prove that neural networks with the hidden-to-output weights calculated by Moore-Penrose generalized inverse of $\mathbf{H}$ (ELM) are universal approximator.

Theorem 2. *For any continuous target function f and N arbitrary distinct samples $(\mathbf{x}_i, \mathbf{t}_i)$, where $\mathbf{x}_i \in \mathbf{R}^n$ and $\mathbf{t}_i \in \mathbf{R}^m$, given*

(i) a standard SLFN with L hidden neurons,
(ii) any $g(\mathbf{a}_i, b_i, \mathbf{x}) \in L^2$ which is infinitely differentiable in any interval, and
(iii) $\mathbf{a}_i$ and b_i randomly chosen from any intervals of $\mathbf{R}^n$ and $\mathbf{R}$, respectively, according to any continuous probability distribution,

then neural networks with the hidden-to-output weights calculated by Moore-Penrose generalized inverse, i.e., ELM, are universal approximator.

Proof. The whole proof consists of three parts: i) we will illustrate an alternative expression of the hidden-to-output weights by orthonormal transformation, which is equivalent to the hidden-to-output weights by Moore-Penrose generalized inverse; ii) we will prove that under the orthonormal condition, the hidden-to-output weights calculated by the equivalent expression can ensure the approximation capability of neural networks; and iii) we use orthonormal transformation to switch linearly independent neurons into orthonormal neurons, and then we can naturally prove that neural networks with ELM are universal approximator.

i) Based on system (4), our aim is to find proper parameters such that

$$\beta_1 g_1(\mathbf{x}_i) + \cdots + \beta_L g_L(\mathbf{x}_i) = f(\mathbf{x}_i), i = 1, \cdots, N \tag{7}$$

where $f(\mathbf{x}_i) = t_i$.
Multiplying equation (7) by $g_j(\mathbf{x}_i)$, for $j = 1, \cdots, L$, and summing the above L equations for i, $i = 1, \cdots, N$, we can obtain

$$\begin{aligned} &\beta_1 \sum_{i=1}^{N} g_1(\mathbf{x}_i) g_j(\mathbf{x}_i) + \cdots + \beta_L \sum_{i=1}^{N} g_L(\mathbf{x}_i) g_j(\mathbf{x}_i) \\ &= \sum_{i=1}^{N} f(\mathbf{x}_i) g_j(\mathbf{x}_i), j = 1, \cdots, L \end{aligned} \tag{8}$$

We can rewrite the above L equations by approx-inner product format (1) as

$$\beta_1 \int_{g_1 g_j} + \cdots + \beta_L \int_{g_L g_j} = \int_{f g_j}, j = 1, \cdots, L \tag{9}$$

which can be condensed as $\widehat{\mathbf{H}}\beta = \widehat{\mathbf{T}}$
where we name $\widehat{\mathbf{H}}$ as approx-inner product hidden layer matrix. Thus, the hidden-to-output weights are the same as the hidden-to-output weights calculated by the inverse of hidden matrix $\mathbf{H}$, i.e.,

$$\mathbf{H}^{\dagger}\mathbf{T} = \widehat{\mathbf{H}}^{-1}\widehat{\mathbf{T}} \tag{10}$$

ii) Approximation under orthonormal condition
Assuming that $\{g_j(\mathbf{x})\}_{j=1}^{L}$ are orthonormal to each other and denoting the actual output of the neural network with L hidden neurons as $f_L(\mathbf{x}) = \sum_{j=1}^{L} \alpha_j g_j(\mathbf{x})$, where $\alpha = [\alpha_1, \alpha_2, \cdots, \alpha_L]^T = \widehat{\mathbf{H}}^{-1}\widehat{\mathbf{T}}$, we have

$$\begin{aligned} &\|f(\mathbf{x}) - f_L(\mathbf{x})\| \\ &\leq \|f(\mathbf{x}) - \sum_{j=1}^{L} \int_{fg_j} g_j(\mathbf{x})\| + \sum_{j=1}^{L} |\int_{fg_j} - \widehat{\int}_{fg_j}| \cdot \|g_j(\mathbf{x})\| \\ &\quad + \|\sum_{j=1}^{L} \widehat{\int}_{fg_j} g_j(\mathbf{x}) - f_L(\mathbf{x})\| \end{aligned} \tag{11}$$

According to Theorem 1, since $\{g_j\}_{j=1}^{L}$ are orthonormal, we know that for any small $\varepsilon > 0$, there exist a L_1, when the neurons' number $L > L_1$, we have

$$\|f(\mathbf{x}) - \sum_{j=1}^{L} \int_{fg_j} g_j(\mathbf{x})\| < \frac{\varepsilon}{3} \tag{12}$$

According to equation (2), there exist a N_1 such that when the samples' number $N > N_1$, we can obtain

$$|\int_{fg_j} - \widehat{\int}_{fg_j}| < \frac{\varepsilon}{3L}$$

Since $\{g_j\}_{j=1}^{L}$ are orthonormal, i.e., $\|g_j(\mathbf{x})\| = 1$, for $j = 1, \cdots, L$, the second item in equation (11) can be expressed as

$$\sum_{j=1}^{L} |\int_{fg_j} - \widehat{\int}_{fg_j}| \cdot \|g_j(\mathbf{x})\| < \frac{\varepsilon}{3L} \cdot \sum_{j=1}^{L} \|g_j(\mathbf{x})\| = \frac{\varepsilon}{3L} \cdot L = \frac{\varepsilon}{3} \tag{13}$$

For the third term in equation (11), we know that $|\int_{fg_j}| < \infty$, so when the number of samples is large enough, $\widehat{\int}_{fg_j}$ approximates $\int_{fg_j}$, thus $|\widehat{\int}_{fg_j}|$ are also bounded. Here we set M_1 as their upper bound, so we have $\|\widehat{\mathbf{T}}\|_1 < L\|\widehat{\mathbf{T}}\|_\infty = L\max_{i=1}^{L}\{|\widehat{\int}_{fg_j}|\} \leq LM_1$.
Since $\{g_k(\mathbf{x})\}_{k=1}^{L}$ are orthonormal each other, when the number of samples is large enough, we have $\widehat{\mathbf{H}} = \mathbf{I}$, where the matrix $\mathbf{I}$ means an identity matrix. In another word, there exists a N_2, when the number of the hidden neurons $N > N_2$, we have

$$\|\mathbf{I} - \widehat{\mathbf{H}}^{-1}\|_1 < \frac{\varepsilon}{3LM_1} \tag{14}$$

Thus, according to Lemma 21 and equation 14, the third term in equation (11) becomes

$$\begin{aligned} &\|\sum_{j=1}^{L} \widehat{\int}_{fg_j} g_j(\mathbf{x}) - f_L(\mathbf{x})\| \leq \sum_{j=1}^{L} |\widehat{\int}_{fg_j} - \alpha_j| \cdot \|g_j(\mathbf{x})\| = \|\widehat{\mathbf{T}} - \alpha\|_1 \\ &\leq \|\mathbf{I} - \widehat{\mathbf{H}}^{-1}\|_1 \cdot \|\widehat{\mathbf{T}}\|_1 < \frac{\varepsilon}{3LM_1} \cdot LM_1 = \frac{\varepsilon}{3} \end{aligned} \tag{15}$$

With equations (12), (13) and (15), for any $\varepsilon > 0$, there exists a L_1, when the number of the hidden neurons $L > L_1$ and the sample number $N > max\{N_1, N_2\}$, equation (11) can be modified as

$$\|f(\mathbf{x}) - f_L(\mathbf{x})\| < \frac{\varepsilon}{3} + \frac{\varepsilon}{3} + \frac{\varepsilon}{3} = \varepsilon \tag{16}$$

According to equation (10), the whole linear system has a unique solution, and then neural networks with the hidden-to-output weights $\beta = \mathbf{H}^{\dagger}\mathbf{T}$ are universal approximator under the orthonormal condition.

iii) Next, we will introduce how to construct orthonormal neurons.
Following the same proof in [15, 7], we can easily obtain that for any infinitely differentiable activation function $g(\cdot) \in L^2$, if $\mathbf{a}_i$ and b_i randomly chosen from any intervals of $\mathbf{R}^n$ and $\mathbf{R}$ based on any continuous probability distribution, the corresponding neurons $\{g_1, g_2, \cdots, g_L\}$ are linearly independent. For those linearly independent neurons $\{g_1, g_2, \cdots, g_L\}$, by applying the standard Gram-Schmidt transformation [12], the sequence $\{g_1(\mathbf{x}), g_2(\mathbf{x}), \cdots, g_L(\mathbf{x})\}$ is transformed as an orthonormal set of basis functions $\{u_1(\mathbf{x}), u_2(\mathbf{x}), \cdots, u_L(\mathbf{x})\}$, i.e.,

$$[u_1(\mathbf{x}), u_2(\mathbf{x}), \cdots, u_L(\mathbf{x})] = [g_1(\mathbf{x}), g_2(\mathbf{x}), \cdots, g_L(\mathbf{x})] \cdot \mathbf{V} \tag{17}$$

where $\mathbf{V}$ is an upper triangular matrix [9].
Denoting

$$\mathbf{U} = \begin{bmatrix} u_1(\mathbf{x}_1) & \cdots & u_L(\mathbf{x}_1) \\ \vdots & \ddots & \vdots \\ u_1(\mathbf{x}_N) & \cdots & u_L(\mathbf{x}_N) \end{bmatrix}_{N \times L}$$

Since $\{u_1(\mathbf{x}), u_2(\mathbf{x}), \cdots, u_L(\mathbf{x})\}$ are orthonormal each other, the weights $\alpha = \mathbf{U}^{\dagger}\mathbf{T}$ can ensure the approximation capability of neural networks, i.e.,

$$\|f(\mathbf{x}) - \sum_{i=1}^{L} \alpha_i u_i(\mathbf{x})\| < \varepsilon. \text{ (c.f. equation (16))} \tag{18}$$

Since $\beta = \mathbf{H}^{\dagger}\mathbf{T}$, we have $\alpha = \mathbf{U}^{\dagger}\mathbf{T} = \mathbf{U}^{\dagger}\mathbf{H}\beta$, i.e.

$$\beta = \mathbf{H}^{\dagger}\mathbf{U}\alpha \tag{19}$$

Denote the actual output of the neural networks is $f_L(\mathbf{x}) = \sum_{k=1}^{L} \beta_k g_k(\mathbf{x})$
According to equations (17) and (19), we have $\mathbf{U} = \mathbf{H}\mathbf{V}$, thus

$$\begin{aligned} & \|f(\mathbf{x}) - f_L(\mathbf{x})\| \\ = & \|f(\mathbf{x}) - \sum_{i=1}^{L} \beta_i g_i(\mathbf{x})\| = \|f(\mathbf{x}) - [u_1(\mathbf{x}), \cdots, u_L(\mathbf{x})]\, \mathbf{V}^{\dagger}\beta\| \\ = & \|f(\mathbf{x}) - [u_1(\mathbf{x}), \cdots, u_L(\mathbf{x})]\, \mathbf{V}^{\dagger}\mathbf{H}^{\dagger}\mathbf{U}\alpha\| \\ = & \|f(\mathbf{x}) - [u_1(\mathbf{x}), u_2(\mathbf{x}), \cdots, u_L(\mathbf{x})]\, \alpha\| = \|f(\mathbf{x}) - \sum_{i=1}^{L} \alpha_i u_i(\mathbf{x})\| \end{aligned} \tag{20}$$

Based on equation (18), we can obtain

$$\|f(\mathbf{x}) - f_L(\mathbf{x})\| = \|f - \sum_{i=1}^{L} \alpha_i u_i(\mathbf{x})\| < \varepsilon$$

4 Performance Evaluation

The main purpose of this paper is to illustrate the universal approximation of ELM and the equivalent relationship between ELM and the orthonormal neural networks, thus we may only need to compare ELM with the orthonormal neural networks in this paper. The simulation results between ELM and the orthonormal neural networks based on many benchmark classification applications will verify our conclusion: neural networks without orthonormal transformation, i.e., ELM, achieve faster training speed under the same generalization performance.

All the benchmark classification simulations are from UCI datasets [16]. For simplicity, all the input data are normalized into the range $[-1,1]$. Neural networks with ELM and with Gram-Schmidt orthonormalization are both assigned the same of number of hidden neurons, i.e. 30 neurons. All the simulations are run in MATLAB 6.5 environment and the same PC with Pentium 4 3.0 GHz CPU and 1G RAM. The activation function used in our proposed algorithms is a simple Gaussian kernel function $g(\mathbf{x}) = \exp(-\gamma\|\mathbf{x}-\mu\|^2)$, the centers μ is randomly chosen from the range $[-1,1]$ whereas the impact factor γ is chosen from the range $(0,0.5)$.

4.1 Benchmark with Classification Problem

Real-world benchmark classification problems are analyzed in this part. The characteristics of the classification datasets are given in Table 4.1. All the training data are also normalized to $[-1,1]$. For each case, the experimental results are based on 50 trials.

The sizes of these datasets are from hundreds to thousands, with both two classes and multiple classes problems. For both ELM and Gram-Schmidt orthonormal neural networks we use a neuron number of 30.

Table 4.2 shows the training and testing accuracy of ELM and Gram-Schmidt orthonormal neural networks with 30 hidden neurons each. As shown in Table 4.2, both the two neural networks achieve good classification accuracy with the same precision, which verifies our equivalent relationship.

Table 4.1. Specification of Benchmark Datasets

Name	No. of Data		Attributes	Classes
	Training	Testing		
BCW	350	349	9	2
German	500	500	24	2
Heart	135	135	13	2
Ionosphere	175	176	33	2
Letter	10000	10000	16	26
Liver	170	175	6	2
Page	2700	2773	10	2
Pima	380	388	8	2
Satellite	4400	2035	36	6
Spambase	3000	1601	57	2
Vehicle	420	426	18	4

Table 4.2. Comparison of Average Accuracy

Name	ELM		Gram-Schmidt	
	Training	Testing	Training	Testing
BCW	97.37	96.30	97.29	96.27
German	74.93	71.94	74.40	71.58
Heart	87.82	80.19	89.38	78.98
Ionosphere	90.01	84.79	89.29	82.89
Letter	59.86	58.82	59.73	59.01
Liver	78.39	69.52	77.41	69.61
Page	94.51	94.16	94.67	94.11
Pima	79.85	76.16	80.11	75.88
Satellite	84.95	84.26	83.19	82.74
Spambase	81.24	81.08	72.51	72.59
Vehicle	78.13	72.50	78.96	72.77

4.2 Face Detection Applications

The analysis and comparison about AdaBoost have been beyond our paper scope. We aim to combine ELM with AdaBoost to improve the face detection accuracy meanwhile maintaining the real-time speed.

First, AdaBoost of Viola-Jones is implemented by using training examples, where positive face images are coming from the UMIST face database and negative background from http://face.urtho.net/, and then our profile face set is used to test the algorithm. As we know, face-like images will pass every stage in AdaBoost, thus it cannot distinguish face-like images from the face images. Next, we use ELM as a filter to exclude those face-like images. Fig. 1(a) shows the detection results of AdaBoost of Viola-Jones, in which we can find that if we only use Adaboost, the final detection results include many false or face-like results. Fig. 1(b) illustrates the detection results of combining AdaBoost of Viola-Jones and ELM. We conclude that through the ELM filter, we can remove those face-like images to improve the face detection accuracy.

(a) (b)

Fig. 1. (a) AdaBoost without ELM, (b) AdaBoost with ELM

5 Conclusions

This paper has successfully proven that neural networks with ELM are universal approximator. The proof also illustrates the equivalent relationship between neural networks without orthonormalization (ELM) and orthonormal neural networks, therefore ELM without orthonormal transformation can achieve a faster learning speed under the same generalization performance. A large number of benchmark classification problems support the conclusion. Due to the fast learning characteristic of ELM, we successfully combine ELM with the real-time AdaBoost algorithm in face detection applications such that the whole system not only retains a real-time learning speed, but also achieves higher face detection accuracy.

Acknowledgments

This research is supported by the Singapore National Research Foundation, Interactive and Digital Media R&D Program, under research grant R-705-000-017-279.

References

1. Yang, M.H., Kriegman, D.J., Member, S., Ahuja, N.: Detecting faces in images: A survey. IEEE Transactions on Pattern Analysis and Machine Intelligence 24(1), 34–58 (2002)
2. Pan, Y., Ge, S.S., Mamun, A.A.: Weighted locally linear embedding for dimension reduction. Pattern Recognitioin 42, 798–811 (2009)
3. Viola, P., Jones, M.: Rapid object detection using a boosted cascade of simple features. In: Proceeding of IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2001), Kauai Marriott, Hawaii, US, December 9-14, pp. 511–518 (2001)
4. Viola, P., Jones, M.: Robust real-time face detection. International Journal of Computer Vision 57(2), 137–154 (2004)
5. Hornik, K.: Approximation capabilities of multilayer feedforward networks. Neural Networks 4, 251–257 (1991)
6. Ito, Y.: Approximation of continuous functions on $\mathbf{R}^d$ by linear combinations of shifted rotations of a sigmoid function with and without scaling. Neural Networks 5, 105–115 (1992)
7. Huang, G.B., Zhu, Q.Y., Siew, C.K.: Extreme learning machine: Theory and applications. Neurocomputing 70, 489–501 (2006)
8. Royden, H.L.: Real Analysis, 3rd edn. Macmillan Publishing Company, New York (1988)
9. Kaminski, W., Strumillo, P.: Kernel orthonormalization in radial basis function neural networks. IEEE Transactions on Neural Networks 8(5), 1177–1183 (1997)
10. Ortega, J.M.: Matrix theory. Plenum Press, New York (1987)
11. Lancaster, P., Tismenetsky, M.: The theory of matrices. Academic Press, London (1984)
12. Stein, E.M., Shakarchi, R.: Real Analysis: Measure Theory, Integration, and Hilbert Spaces. Princeton University Press, Princeton (2005)
13. Serre, D.: Matrices: Theory and Applications. Springer, New York (2002)
14. Rao, C.R., Mitra, S.K.: Generalized Inverse of Matrices and Its applications. John Wiley & Sons, Inc., New York (1971)
15. Tamura, S., Tateishi, M.: Capabilities of a four-layer feedforward neural networks: Four layers versus three. IEEE Transactions on Neural Network 8(2), 251–255 (1997)
16. Blake, C., Merz, C.: UCI repository of machine learning databases. Department of Information and Computer Sciences, University of California, Irvine, USA (1998), `http://www.ics.uci.edu/~mlearn/MLRepository.html`

Determination of Gender and Age Based on Pattern of Human Motion Using AdaBoost Algorithms

Santoso Handri[1], Shusaku Nomura[1], and Kazuo Nakamura[2]

[1] Top Runner Incubation Center for Academia-Industry Fusion
[2] Department of Management and Information Systems Science
Nagaoka University of Technology, Nagaoka, Niigata – Japan
handri.santoso@mis.nagaokaut.ac.jp,
{nakamura,nomura}@kjs.nagaokaut.ac.jp

Abstract. Automated human identification by their walking behavior is a challenge attracting much interest among machine vision researchers. However, practical systems for such identification remain to be developed. In this study, a machine learning approach to understand human behavior based on motion imagery was proposed as the basis for developing pedestrian safety information systems. At the front end, image and video processing was performed to separate foreground from background images. Shape-width was then analyzed using 2D discrete wavelet transformation to extract human motion features. Finally, an adaptive boosting (AdaBoost) algorithm was performed to classify human gender and age into its class. The results demonstrated capability of the proposed systems to classify gender and age highly accurately.

Keywords: human motion, gender and age, CHIAN, and competitive learning algorithm.

1 Introduction

Much research has gone into improving pedestrian safety by addressing street crossing problems [1-4]. Designing, engineering, operating, and maintaining roadways and pedestrian facilities have been developed to make better pedestrian-friendly environment. Among engineering solutions developed to do so the intelligent transportation systems (ITS) related to pedestrian safety and road access targets problems pedestrian face in crossing streets. These include increasing motorist awareness of pedestrians, giving feedback to waiting and street-crossing pedestrians, implementing pedestrian detection by motorists, and addressing visual impairment issues. Although, pedestrian-crossing facilities now detect pedestrian movement to be feedback to systems adjusting the parameters, the detection of individual pedestrians remains to be developed.

Sensor technology emulating human vision such as video cameras could help solve these problems. Surveillance cameras are installed in many public areas and computer-based image processing is promising as a way to handle vast amounts of camera-generated image data. One objective of integrated surveillance is to warn system operators when events that may require human intervention, such as avoiding

J.-H. Kim et al. (Eds.): FIRA 2009, LNCS 5744, pp. 311–319, 2009.

accidents [5], or preventing vandalism, are detected. Such warnings are reliable, however, only if systems detect and determine pedestrian behavior. Automatically monitoring pedestrian behavior via computers could conceivably much work in continuously monitoring and operating the surveillance systems.

Similar to the uniqueness of fingerprints or faces, ways of walking are believed to be unique to the individual. Psychological studies have sought to determine the human ability to recognize persons based on their gait [6]. A new area in computer vision is the automated analysis of gait from motion imagery and attempts have been made to use gait as an identification parameter in biometrics, medical rehabilitation and sports, etc. Little works has been done, however, in pursuing pedestrian safety based on their motion.

Walking is a daily activity, and it is one of most basic human actions. It can be observed regardless of public and private matters through all of space-time dimension. By seeing the human walking behavior, the various perceptions of body conditions can be easily observed when abnormality is found in limbs, in human motor coordination, or in human sensory function. The pedestrian posture and walking speed condition in which the defective function is found, are reflected from those scenes.

In this research, space-time pattern information was extracted based on the dynamic scene acquired from a video image. The walking condition of person which might represent the perception of body modality differences between age- and gender-groups was evaluated from the dynamic features of walking pattern. In this study, an intelligent information processing mechanism that embedding the aspect of the human thinking way connected with the factor of the walking appearance was determined by the soft computing approach. This system is expected to be capable of detecting specific objects in the street based on their behavior attributes such as gait. If we are able to implement this research in real world condition, i.e., determining normal and abnormal pattern of human motion, it will be helpful in upgrading the safety of pedestrians and decrease the occurrence of untoward incidents.

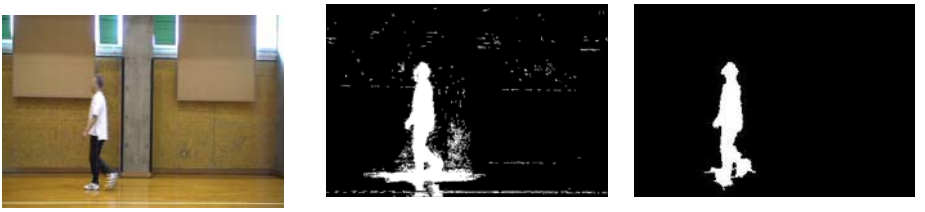
2 Preprocessing of Human Motion Imagery

2.1 Extraction of Human Motion Features

The segmenting of moving objects in image sequences is a fundamental in surveillance application steps. Background segmentation involves calculating reference images, subtracting new frames from reference images, thresholding the result, and updating reference images. Here, Gaussian distribution of adaptive mixture Gaussian models proposed by Stauffer *et al.* was employed to separate foreground and background images [7].

After this separation, pixels containing pictures of pedestrians and other moving objects are detected in binary image. The detected image can be distorted by noise introduced during image acquisition or by natural causes such as shadows. Hue, Saturation and Value (HSV) color space corresponds closely to human color perception and has been proven more accurate in distinguishing shadows than red, green, and blue (RGB) color space or gray-scale space [8]. An attempt is made to determine how occlusion due to shadow changes H, S and V values. Morphological

filter such as dilation and erosion is then employed to remove anomalies from the detected region, such as spurious pixels, and holes in object, yielding the foreground image result as shown in Fig. 1.



(a) (b) (c)

Fig. 1. Object detection. (a) Original video frame. (b) Detected foreground image (c) Eliminating shadow and spurious pixels at detected object.

Once the spatial silhouette of a walking object is extracted, its boundary is obtained easily using a border-following algorithm based on connectivity. An appearance-based approach is used to extract basic image-motion features; similar to that proposed by Kale *et al.* [9], i.e., the width of the contour of a silhouette is extracted as motion features as shown in Fig. 2. Image silhouette widths is calculated as the difference in locations of left and right-most boundary pixels for rows and a width vector generated for frames.

Silhouette contour widths contain structural and dynamic aspects and represent patterns of human motion compactly. Spatial-temporal widths plotted in Fig. 3 show that the width vector is roughly regular and gives the extent of movement of different parts of the body. Most such changes occur in the arms and legs during walking. Hence, one would intuitively expect that variations of the widths vector might correlate with the periodic of human walking. Feature transformation from the space-time domain to the frequency domain thus is proposed to extract human motion features in this study.

Wavelets are functions that satisfy certain mathematical requirements and are used in representing data or other functions. This idea is based on Fourier theorem that any periodic signal can be expressed in sine and cosine term [10]. However, in wavelet analysis, the *scale* that we use to look at data plays a special role. Wavelet algorithms process data at different *scales* or *resolutions.* If we look at a signal with a large "window" we would notice gross features. Similarly, if we look at a signal with a small "window" we would notice small features. The result in wavelet analysis is to see both the forest and the trees, so to speak. This makes wavelets interesting and useful. For many decades, scientists have wanted more appropriate functions than the sine and cosine which comprise the bases of Fourier analysis. By their definition, these functions are non-local (and stretch out to infinity). They therefore do a very poor job in approximating sharp spikes. But with wavelet analysis, we can use approximating functions that are contained neatly in finite domains. Wavelets are well-suited for approximating data with sharp discontinuities.

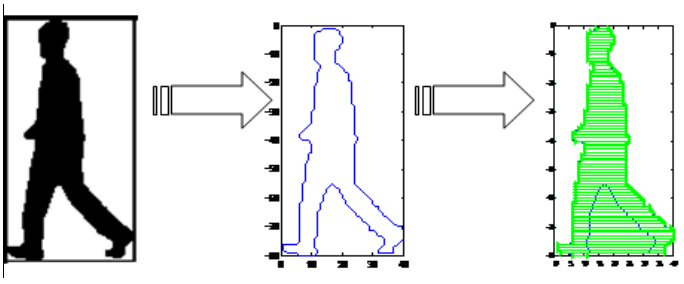


Fig. 2. Extraction of silhouette width

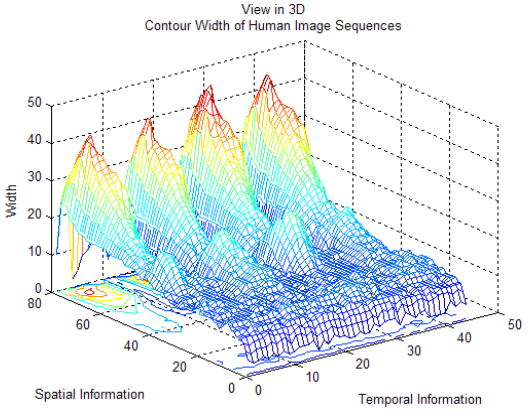


Fig. 3. Spatiotemporal width of human motion

The wavelet analysis procedure is to adopt a wavelet prototype function, called an *analyzing wavelet* or *mother wavelet*. Temporal analysis is performed with a contracted, high-frequency version of the prototype wavelet, while frequency analysis is performed with a dilated, low-frequency version of the same wavelet. Because the original signal or function can be represented in terms of a wavelet expansion using coefficients in a linear combination of the wavelet functions, data operations can be performed using just the corresponding wavelet coefficient.

2.2 Selection of Human Motion Features

In practical use, not all extracted features either need to be, or indeed, can be used, because an increasing number of features may actually reduce performance of classification. Therefore, it is necessary to remove irrelevant features to ensure efficiency recognition accuracy and interpretation purposing ease. The process of removing irrelevant and redundant features is called feature selection [11]. A limited, yet salient, feature set simplifies both pattern representation and classifiers built in selected representation. Consequently, the resulting classifier will be faster and use less memory. For that reason, we used feature selection to select a yet smaller subset of extracted features deemed to be the most distinguishing and informative.

Feature subset selection (FSS) is the preprocessing part of the model that selects useful features for classification. Selection is based on the individual advantages of each feature. The T-Test criterion measures individual feature significance based on ranking features $\boldsymbol{x}$ using an independent evaluation criterion for binary classification. The D-dimensional input vector is denoted as $x=(x_1, x_2, \cdots, x_D)$, where the number of examples belonging to the effective group is n_{+1}, the examples belonging to the ineffective group n_{-1}, the mean of the j-th feature of the effective group $\mu_{j,+1}$, the mean of the j-th feature of ineffective group $\mu_{j,-1}$, and their standard deviations $\sigma_{j,+1}$ and $\sigma_{j,-1}$. The significance of each feature x_j is measured as follows:

$$F(x_j)=\left|\frac{(\mu_{j,+1}-\mu_{j,-1})}{\sqrt{\left(\frac{\sigma_{j,+1}}{n_{j,+1}}+\frac{\sigma_{j,-1}}{n_{j,-1}}\right)}}\right| \tag{1}$$

This criterion is interpreted as finding the one single feature that best discriminates among both groups in feature space. The greater this score, the better the feature's discrimination. Based on this score, individual features are assigned by rank of significance. Features are selected using a certain number of features from the top. Feature subset selection is effective if the condition of variables is independent.

3 AdaBoost Algorithms Approach to Human Attributes

3.1 AdaBoost Algoritms

The AdaBoost algorithm, introduced in 1995 by Freund and Schapire [12], is used to boost the classification performance of a simple learning algorithm. It does this by combining a collection of weak classification functions to form a stronger classifier. In the language of boosting the simple learning algorithm is called a weak leaner. Pseudo-code for AdaBoost is given as:

- Given $(x_1, y_1), \cdots, (x_m, y_m)$
- where $x_i \in \mathrm{X}, y_i \in \mathrm{Y} = \{-1, +1\}$
- Initialize $D_1(i) = \frac{1}{m}$
- for $\mathrm{t} = 1, \cdots, \mathrm{T}$
  - Train weak learner using distribution $\mathrm{D_t}$
  - Get the weak hypothesis $\mathrm{h_t}: \mathrm{X} \rightarrow \{-1, +1\}$ with error
    $$\varepsilon_t = Pr_{i-D_t}[h_t(x_i) \neq y_i]$$
  - Choose
    $$\alpha = \frac{1}{2} ln\left(\frac{1-\varepsilon_t}{\varepsilon_t}\right)$$

- Update

$$D_{t+1} = \frac{D_t(i)}{Z_t} \times \begin{cases} e^{-\alpha t} \; if \; h_t(x_i) \neq y_i \\ e^{\alpha t} \; if \; h_t(x_i) = y_i \end{cases}$$

$$= \frac{D_t(i) exp\left(-\propto_t y_i h_t(x_i)\right)}{Z_t}$$

- Where Z_t is a normalization factor (chosen so that D_{t+1} will be a distribution).
- Output the final hypothesis :

$$H(x) = sign\left(\sum_{t=1}^{T} \propto_t h_t(x)\right)$$

The algorithm takes as input a training set $(x_1; y_1),\ldots, (x_m; y_m)$ where each x_i belongs to some *domain* or *instance space* X, and each *label* y_i is in some label set Y . For most of this paper, we assume Y = {-1, +1}. AdaBoost calls a given *weak* or *base learning algorithm* repeatedly in a series of rounds t = 1,…, T. One of the main ideas of the algorithm is to maintain a distribution or set of weights over the training set. The weight of this distribution on training example i on round t is denoted $D_t(i)$. Initially, all weights are set equally, but on each round, the weights of incorrectly classified examples are increased so that the weak learner is forced to focus on the hard examples in the training set.

3.2 The Verification Methods

In this paper, the classification accuracy of Gentle and Modest AdaBoost was tested by the leave-one-out cross validation (LOO-CV) method [13], which can be applied when the samples are small. The procedure consists of picking up one example for testing while the rest of the data are used to train the classifiers, and then testing the removed example. After testing, the classification result is recorded. The process is repeated until all examples have been tested. The accuracy of classifier is evaluated by calculating the average error as following;

$$\bar{e} = \frac{1}{N} \sum_{i}^{N} e_i \qquad (2)$$

where N is the total number of experiments conducted.

4 The Experiments and Results

Videos of human walking were taken in the Gymnasium of the Nagaoka University of Technology and Nagaoka Long Life Center (LLC) during spring, March 2008. The samples consist of adult and senior citizens. There are 20 adults, 6 females and 14 males. While for the senior citizens, there are 18 females and 15 males. There are totally 53 persons who joined this experiment as shown in Table 1. The age categorization is set as elderly if age of participants is more than 50 year old. It is categorized as adult if the age is less than 40 year old. An image sequence in the database contains only a single subject walking at normal speed and was acquired at 30 fps with 720×480 colour pixels from good quality progressive scan hard-disk

Table 1. Participant Profiles

	Male	Female	Total
Young	14	6	20
Elderly	15	18	33
	29	24	53

Table 2. Age classification results

Classification	AdaBoost Algorithms	
Type	Gentle	Modest
Features Used	Accuracy (%)	
10	83.0%	79.2%
20	69.8%	83.0%
30	79.2%	83.0%
40	84.9%	88.7%
Average	79.2%	83.5%

Table 3. Gender classification results

Classification	AdaBoost Algorithms	
Type	Gentle	Modest
Features Used	Accuracy (%)	
10	81.1%	90.6%
20	75.5%	79.2%
30	73.6%	83.0%
40	73.6%	86.8%
Average	75.9%	84.9%

cameras Sony DCS-SR100. All subjects in the database are filmed at an oblique angle. Each subject has at least four image sequences and each image sequence contains at least one gait cycle, together with the background and other supporting data. The human body extraction from the image sequences can be achieved through the background subtraction. After that, the shadow detection and the elimination of spurious pixel were performed for smoothing the detected object. Finally, the body contour is extracted by using a border following algorithm based on connectivity.

Shape width was analyzed by 2D discrete wavelet transformation to extract human movement features. The features extracted were then selected to get features effective in classification using the t-Test criterion. Several top-ranked features were used for classification, i.e., 10, 20, 30, and 40 features. Two types of AdaBoost classifiers, i.e., Gentle and Modest AdaBoost algorithms were carried out for age and gender classification Furthermore; the network was trained until the error is below the

threshold value or until the maximum epoch is reached. Once the weights have been determined, the network can be used as a classifier. The classification accuracy was performed by measuring the average error of each classifier based on leave-one-out cross validation (LOO-CV) method. The experimental results are summarized in Table 2 and Table 3.

In Table 2, age classification, performed by Modest AdaBoost algorithms, showed higher accuracy result comparing with Gentle AdaBoost algorithms in average. The effective features used for age classification, performed by Gentle and Modest algorithms, were 40 features.

Table 3 also showed that the Modest AdaBoost algorithms outperformed the Gentle AdaBoost algorithms for gender classification. For this classification, 10 effective features gave better performance than others.

It was concluded that feature extraction focusing on silhouette width may more simply classify gender than age. To improve age classification, it may be better to separate gender among adult and older persons first before classifying age. Such hierarchical processing may yield higher accuracy in pedestrian attribute classification.

However the confirmations of those hypotheses have to be proofed by taking several testing for supporting this evidence in future research.

5 Conclusion

This study has described automated age and gender classification using computer vision and machine learning to extract shape-motion by discrete wavelet transform based on spatiotemporal information. After shape widths was analyzed using 2D wavelet transform to extract movement features, two types of AdaBoost algorithms, Gentle and Modest were used to analyze age and gender discrimination based on extracted features. Modest AdaBoost algorithm has produced average age classification of 83.5% and gender classification of 84.9.5%, compared to a Gentle AdaBoost algorithm performance of 79.2% for age classification and 75.9% for gender classification. Modest AdaBoost algorithms performance is thus higher than Gentle AdaBoost algorithm in age classification and in gender classification problem.

References

[1] Pedestrian and Bicycle Information Center, http://www.walkinginfo.org
[2] Hummel, T.: Dutch Pedestrian Safety Research Review. Tech. report FHWA-RD-99-092, SWOV Institute for Road Safety Research (December 1999)
[3] Banerjee, I., Shladover, S.E., Misener, J.A., Chan, C.Y., Ragland, D.R.: Impact of Pedestrian Presence on Movement of Left-Turning Vehicles: Method, Preliminary Research & Possible Use in Intersection Decision Support, University of California, Berkeley (2004)
[4] Bechtel, A.K., Geyer, J., Ragland, D.R.: A Review of ITS-based Pedestrian Injury Countermeasures. University of California, Berkeley (2003)

[5] Kamijo, S., Matsushita, Y., Ikeuchi, K., Sakauchi, M.: Traffic Monitoring and Accident Detection at Intersections. IEEE Transactions on Intelligent Transportation Systems 1(2), 108–118 (2000)
[6] Nixon, M.S., Carter, J.N., Grant, M.G., Gordon, L., Hayfon-Acquah, J.B.: Automatic Recognition by Gait: Progress and Prospects. Sensor Review 23(4), 323–331 (2003)
[7] Stauffer, C., Grimson, W.E.L.: Learning patterns of activity using real-time tracking. IEEE Transaction on Pattern Analysis & Machine Intelligence 22(8), 747–757 (2000)
[8] Prati, A., Mikic, I., Grana, C., Trivedi, M.M.: Shadow Detection Algorithms for Traffic Flow Analysis: a Comparative Study. In: IEEE Int'l. Conf. On Intelligent Transportation Systems, Oakland, California (August 2001) (submitted)
[9] Kale, A., Cuntoor, N., Rajagopalan, A.N., Yegnanarayana, B., Chellappa, R.: Gait analysis for human identification. In: Proceedings of 3rd International Conference on Audio and Video Based Person Authentication (June 2003)
[10] Polikar, R.: Wavelet Tutorial Part II, http://users.rowan.edu/~polikar/WAVELETS/WTpart2.html
[11] Pavel, P., Novovicova, J.: Novel Methods for feature subset selection with respect to problem domain. In: Feature Extraction, Construction and Selection: A Data Mining Perspective, 2nd edn. Kluwer Academic Publisher, Dordrecht (2001)
[12] Freund, Y., Schapire, R.E.: A decision-theoretic generalization of on-line learning and an application to boosting. Journal of Computer and System Science 55(1), 119–139 (1997)
[13] Cawley, G.C., Talbot, N.L.C.: Efficient leave-one-out cross-validation of kernel fisher discriminant classifiers. Pattern Recognition 36(11), 2585–2592 (2003)

Author Index