LNCS 6574

Katsumi Tanaka
Peter Fröhlich
Kyoung-Sook Kim (Eds.)

Web and Wireless Geographical Information Systems

10th International Symposium, W2GIS 2011
Kyoto, Japan, March 2011
Proceedings

Springer

Lecture Notes in Computer Science 6574

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

Katsumi Tanaka Peter Fröhlich
Kyoung-Sook Kim (Eds.)

Web and Wireless Geographical Information Systems

10th International Symposium, W2GIS 2011
Kyoto, Japan, March 3-4, 2011
Proceedings

Springer

Volume Editors

Katsumi Tanaka
Kyoto University, Graduate School of Informatics
Department of Social Informatics
Yoshida-Honmachi, Sakyo-ku, Kyoto 606-8501, Japan
E-mail: tanaka@dl.kuis.kyoto-u.ac.jp

Peter Fröhlich
Telecommunications Research Center Vienna (ftw.)
Donau-City-Straße 1/3, 1220 Vienna, Austria
E-mail: froehlich@ftw.at

Kyoung-Sook Kim
Knowledge Creating Communication Research Center
National Institute of Information and Communications Technology
3-5 Hikaridai, Seika-cho, Soraku-gun, Kyoto 619-0289, Japan
E-mail: ksookim@nict.go.jp

ISSN 0302-9743 e-ISSN 1611-3349
ISBN 978-3-642-19172-5 e-ISBN 978-3-642-19173-2
DOI 10.1007/978-3-642-19173-2
Springer Heidelberg Dordrecht London New York

Library of Congress Control Number: 2011920518

CR Subject Classification (1998): H.2, H.3, H.4, H.5, C.2

LNCS Sublibrary: SL 3 – Information Systems and Application, incl. Internet/Web and HCI

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India

Printed on acid-free paper

Springer is part of Springer Science+Business Media (www.springer.com)

Preface

Continuing advances in the development of wireless and Internet technologies generate ever increasing interest in the diffusion, usage, and processing of geo-referenced data of all types. Spatially-aware wireless and Internet devices also offer new ways of accessing and analyzing geo-spatial information in both real-world and virtual spaces. Consequently, new challenges and opportunities have been provided that expand the traditional GIS research scope into the realm of intelligent media, including geomedia with context-aware behaviors for self-adaptive use and delivery.

The International Symposium Series on Web and Wireless Geographical Information Systems (W2GIS) provides an up-to-date review of advances on recent development and research results in this research field. W2GIS is a series of successful events alternating locations between East Asia and Europe. The 10th symposium returned to Kyoto, where the W2GIS series was first launched in 2001. This proceedings volume contains the papers selected for presentation at W2GIS 2011, held in March 2011 in Kyoto.

We had 36 submissions from 20 countries, many of them of excellent quality. Each paper received four reviews. A total of 13 full and 3 short papers plus 2 short keynote papers were finally accepted for presentation. The program and proceedings cover a wide range of topics including geographic information retrieval on the web, geo-spatial semantic and sensor web, location-based services, advanced GIS visualization techniques, personalization and adjustment for mobile GIS applications, and geo-spatial data quality and context processing.

We had the privilege of having two invited talks by Steve Coast (founder of OpenStreetMap) and Xing Xie (Microsoft Research Asia) plus one special presentation by Nobuo Kawaguchi from Nagoya University in Japan. The best paper of the symposium was selected by the Steering Committee and invited for publication in the Transactions in GIS.

We wish to thank the authors who contributed to this symposium, for the high quality of their papers and presentations, and the Springer LNCS team, for their support. Furthermore, we are very grateful to the Kyoto University GCOE program, for hosting the event. We would also like to thank all the Program Committee members for the timeliness and quality of their evaluations, as well as the Steering Committee and Local Organization Committee for their continuous support.

March 2011

Katsumi Tanaka
Peter Fröhlich
Kyoung-Sook Kim

W2GIS 2011 Organization

Symposium Chair

Katsumi Tanaka	Kyoto University, Japan

Program Co-chairs

Peter Fröhlich	Forschungszentrum Telekommunikation Wien (FTW), Austria
Kyoung-Sook Kim	National Institute of Information and Communications Technology (NICT), Japan

Local Arrangements Co-chairs

Takafumi Nakanishi	National Institute of Information and Communications Technology (NICT), Japan
Kyoung-Sook Kim	National Institute of Information and Communications Technology (NICT), Japan

Steering Committee

Michela Bertolotto	University College Dublin, Ireland
James D. Carswell	Dublin Institute of Technology, Ireland
Christophe Claramunt	Naval Academy Research Institute, France
Max J. Egenhofer	NCGIA, University of Maine, USA
Ki-Joune Li	Pusan National University, Korea
Taro Tezuka	Ritsumeikan University, Japan
Christelle Vangenot	University of Geneva, Switzerland

Program Committee

Alia I. Abdelmoty	Cardiff University, UK
Alain Bouju	University of La Rochelle, France
Tomas Brinkhoff	IAPG, Germany
Elena Camossi	JRC ISPRA, Ireland
Keith Cheverst	Lancaster University, UK
R.A. de By	ITC, Netherlands
S. Di Martino	Naples University, Italy

Matt Duckham	University of Melbourne, Australia
Keith Gardiner	DIT, Ireland
Jerome Gensel	University of Grenoble, France
Ralf H. Güting	FernUniversität in Hagen, Germany
Yoshiharu Ishikawa	Nagoya University, Japan
Bin Jiang	University of Gavle, Sweden
Hanssan Karimi	University of Pittsburgh, USA
Yutaka Kidawara	NICT, Japan
Barend Köbben	ITC, Netherlands
Jitka Komarkova	Pardubice University, Czech Republic
Yongjin Kwon	Korea Aerospace University, Korea
Ray Larson	UC Berkeley, USA
Duan Li	The Chinese University of Hong Kong, Hong Kong, China
Songnian Li	Ryerson University, Canada
Xing Li	East China Normal University, China
Steve Liang	University of Calgary, Canada
Miguel R. Luaces	University of Coruña, Spain
Gavin McArdle	University College Dublin, Ireland
E. Mac Aoidh	JRC ISPRA, Italy
Herve Martin	University of Grenoble, France
Pedro R. Muro-Medrano	University of Zaragoza, Spain
Dieter Pfoser	Computer Technology Institute, Greece
Cyril Ray	Naval Academy Research Institute, France
Philippe Rigaux	University of Paris-Dauphine, France
Markus Schneider	University of Florida, USA
Johannes Schöning	DFKI, Germany
Cyrus Shahabi	University of Southern California, USA
Kazutoshi Sumiya	University of Hyogo, Japan
Yannis Theodoridis	University of Piraeus, Greece
Martin Tomko	Universtiy of Zurich, Switzerland
Taketoshi Ushiama	Kyushu University, Japan
Stephan Winter	University of Melbourne, Australia
Alexander Zipf	University of Heidelberg, Germany

Sponsors

Global COE (GCOE) Program, Kyoto University

Global COE
知識循環社会のための情報学教育研究拠点
Kyoto University

Table of Contents

Invited Talks

Build Intelligence from the Physical World 1
Xing Xie

How OpenStreetMap Is Changing the World 4
Steve Coast

Geographic Information Retrieval

GeoST: Geographic, Thematic and Temporal Information Retrieval from Heterogeneous Web Data Sources 5
Felix Mata and Christophe Claramunt

Extraction and Geographical Navigation of Important Historical Events in the Web 21
Mitsuo Yamamoto, Yuku Takahashi, Hirotoshi Iwasaki, Satoshi Oyama, Hiroaki Ohshima, and Katsumi Tanaka

Investigation of Recommendation Method Considering Region-Restrictedness of Spots 36
Kenta Oku and Fumio Hattori

Geospatial Semantic and Sensor Web

A Case Study of Participatory Data Transfer for Urban Temperature Monitoring 50
Houtan Shirani-Mehr, Farnoush Banaei-Kashani, Cyrus Shahabi, and Lin Zhang

A Folksonomy-Based Recommendation System for the Sensor Web 64
Rohana Rezel and Steve Liang

Semantically Enriching VGI in Support of Implicit Feedback Analysis 78
Andrea Ballatore and Michela Bertolotto

Location-Based Services and Applications

A Mobile Navigation and Orientation System for Blind Users in a Metrobus Environment 94
Felix Mata, Andres Jaramillo, and Christophe Claramunt

A Rich Cloud Application to Improve Sustainable Mobility 109
Sergio Di Martino, Clemente Giorio, and Raffaele Galiero

Mobile Task Computing: Beyond Location-Based Services and EBooks 124
John Liagouris, Spiros Athanasiou, Alexandros Efentakis, Stefan Pfennigschmidt, Dieter Pfoser, Eleni Tsigka, and Agnès Voisard

Advanced GIS Visualization Techniques

Cinematic Street: Automatic Street View Walk-through System Using Characteristics of Modified Maps 142
Kaori Kobayashi, Daisuke Kitayama, and Kazutoshi Sumiya

HYDROSYS – A Mixed Reality Platform for On-Site Visualization of Environmental Data 159
Antti Nurminen, Ernst Kruijff, and Eduardo Veas

Inferring and Focusing Areas of Interest from GPS Traces 176
Pablo Martinez Lerin, Daisuke Yamamoto, and Naohisa Takahashi

Web and Mobile Map Adaptation

Remote-Evaluation of User Interaction with WebGIS 188
Jens Ingensand and François Golay

Touch2Query Enabled Mobile Devices: A Case Study Using OpenStreetMap and iPhone 203
Junjun Yin and James D. Carswell

Geospatial Data Quality and Context

Delimiting Imprecise Regions with Georeferenced Photos and Land Coverage Data 219
Bruno Martins

Visualization of Relationships among Historical Persons Using Locational Information 230
Takahiko Osaki, Sho Itsubo, Fuminori Kimura, Taro Tezuka, and Akira Maeda

Author Index 241

Build Intelligence from the Physical World

Xing Xie

Microsoft Research Asia
4F, Sigma Center, No.49, Zhichun Rd., Haidian District,
Beijing, P.R. China
xingx@microsoft.com

Abstract. By accumulating and aggregating physical world contextual information from multiple users and multiple devices over a long period, we can obtain collective social intelligence from them. Based on this, more innovative Internet services can be developed to facilitate people's everyday lives. In this talk, I will present our recent work on this direction, as well as other related works in Microsoft and the industry.

Context awareness is a key concept in ubiquitous computing. It means linking changes in the environment with computer systems. In other words, computing systems become more intelligent through analyzing and reacting to the physical world surrounding them. The coming era of cloud computing brings new opportunities to this long studied research area. By accumulating and aggregating physical world contextual information from multiple users and multiple devices over a long period, we can obtain collective social intelligence from them. Based on this, more innovative Internet services can be developed to facilitate people's everyday lives. At Microsoft Research Asia, we are working on various technologies with a view to managing physical world information and building intelligence from them. We link data generated by different people, services or sensors together, based on a unified knowledge model and provide the intelligence as a service in the Cloud. In this talk, I will present our recent work on mining user generated GPS trajectories, geo-tagged photos and Taxi trajectories, as well as other related works in Microsoft and the industry.

The advance of location-acquisition technologies like GPS and Wi-Fi has enabled people to record their location history with a sequence of time-stamped locations, called trajectories. These trajectories imply to some extent users' life interests and preferences, and have facilitated people to do many things, such as online life experience sharing, sports activity analysis and geo-tagging multimedia content. Using these trajectories, we can not only connect locations in the physical world but also bridge the gap between users and locations and further deduce the connections among users. That is, we are able to understand both users and locations based on the trajectories.

In this talk, we will first introduce GeoLife project, which is a social networking service incorporating users, locations and user-generated GPS trajectories. People access a sequence of locations in the real world and generate many trajectories in a form of GPS logs. Based on these GPS trajectories, we conducted three lines of research. 1) Understanding trajectories: For example, we provide a

K. Tanaka, P. Fröhlich, and K.-S. Kim (Eds.): W2GIS 2011, LNCS 6574, pp. 1–3, 2011.

method to search for the trajectories by giving a set of locations of interests, or by issuing a spatial query range combined with a temporal interval as a query. Meanwhile, we infer the transportation modes that a user took when generating a GPS trajectory. 2) Understanding users: First, we estimate the similarity between each pair of users in terms of their location histories represented by GPS trajectories. Second, using an iterative inference model, we compute the travel experience of each user based on their location history, and then find out travel experts in a geo-region by ranking users according to the experiences. Third, we can infer the activities that a user can perform in a location based on multiple users' location histories and the associated comments. 3) Understanding locations: One example is that we mine the top interesting locations and travel sequences in a given region from a large number of users' GPS trajectories. Another example is we predict a user's interests in an unvisited location by involving the GPS trajectories of the user and that of others.

With the advancement of digital camera technology, people can casually take photos when they find something attractive. These photos can be easily shared on photo sharing Web sites, such as Flickr and Picasa. To make photo organization easy, popular photo sharing sites provide functions to assign metadata to photos, i.e., tags and geo-tags. Tags can add descriptions that are understood by humans and geo-tags convey locations where the photos were captured represented by their latitudes/longitudes. The number of geo-tagged photos in Flickr has already exceeded 100 million in 2009 and is growing at an amazingly rapid pace.

Since these geo-tagged photos are associated with locations, we can trace people's trips from them. The second project which we will present in this talk is a system to detect frequent photo trip patterns from shared geo-tagged photos. Photo trip patterns are beneficial for practical applications and services, such as travel advisory. The advantage of trip pattern mining from geo-tagged photos is that these photos are of previous travelers' actual experiences. Therefore, mined trip patterns would be highly practical and may show word-of-mouth information. Another advantage is the scale; we can collect millions of people's data from the Web. We develop such an application where trip patterns are used for travel plan recommendation that provides people with new ideas of trips and the highlights of the places they may visit.

Our method automatically segments each user's photo collection into photo trips as sets of photos associated with cities the user visited, the duration of visit to the cities, and clouds of photo tags. The photo trips are categorized according to their trip themes. Finally, our method mines the photo trips of each trip theme to detect frequent photo trip patterns, as city sequences frequently visited and typical visit durations. To add more information that characterizes the photo trip patterns, e.g., what to see and expect during the trips, we further mine photo tags to extract descriptions of photo trip patterns.

We also try to discover and summarize city landmarks from geo-tagged photos. We design a hierarchical visual-textual clustering scheme to structure city scene photos. Then, to discover and summarize landmarks, on one hand, a PhotoRank algorithm is developed to discover representative views within each scene; on the other hand, a Landmark-HITS model is presented to integrate community, context, content cues in representative scene highlighting. Finally, to achieve personalized tourist recommendation, we adopt a collaborative filtering scheme which leveraging blog community knowledge to predict further preferences of target users.

Finding efficient driving directions has become a daily activity and been implemented as a key feature in many map services. A fast driving route saves not only the time of a driver but also energy consumption. In practice, big cities with serious traffic problems usually have a large number of taxis traversing on road surfaces. For the sake of management and security, these taxis have already been embedded with a GPS sensor, which enables a taxi to report on its present location to a data center in a certain frequency. Thus, a large number of time-stamped GPS trajectories of taxis have been accumulated and are easy to obtain. Intuitively, taxi drivers are experienced drivers who can usually find out the fastest route to send passengers to a destination based on their knowledge. When selecting driving directions, besides the distance of a route, they also consider other factors, such as the time-variant traffic flows on road surfaces, traffic signals and direction changes contained in a route, as well as the probability of accidents. These factors can be learned by experienced drivers but are too subtle and difficult to incorporate into existing routing engines. Therefore, these historical taxi trajectories, which imply the intelligence of experienced drivers, provide us with a valuable resource to learn practically fast driving directions.

In the end of this talk, we will describe T-Drive project, which mines smart driving directions from a large number of real-world historical GPS trajectories of taxis. Taxi trajectories are aggregated and mined in the Cloud to answer queries from ordinary drivers or Internet users. Given a start point and destination, our method can suggest the practically fastest route to a user according to his/her departure time and based on the intelligence mined from the historical taxi trajectories. As the taxi trajectories are constantly updated in the Cloud, the suggested routes are state-of-the-art. In our approach, we model a large number of historical taxi trajectories with a time-dependent landmark graph, in which a node represents a road segment frequently traversed by taxis. Based on this landmark graph, we perform a two-stage routing algorithm that first searches the landmark graph for a rough route and then finds a refined route sequentially connecting these landmarks.

How OpenStreetMap Is Changing the World

Steve Coast

Microsoft
One Microsoft way, Redmond, WA 98052, USA
steve@asklater.com

Abstract. This talk reviews these and many more aspects of OpenStreetMap and gives an in depth feel of the past, present and future of the project.

OpenStreetMap is a map of the world built by volunteers. Using a peer production model similar to wikipedia approximately 300,000 people collaboratively map features such as roads, parks and buildings and submit them to be shared. That map data is made available under an open license for anybody to use as they see fit, with two conditions. First that if you use the maps you must say where they came from and second that you release any changes to the maps under the same creative commons license, CCBYSA.

The project was founded in 2004 as a practical solution to the lack of freely available map data. Map data is traditionally collected and distributed by private enterprises and governments which tend to charge high prices and restrict the usage rights of the data carefully, much as encyclopedia britannica does in contrast to the open wikipedia model. OpenStreetMap is very similar in that regard.

Data is collected in an ad hoc and undirected manner by volunteers worldwide. These volunteers use an open tagging system to classify data as they see fit, instead of a hierarchical top-down ontological system. This leads to man benefits such as allowing mallets to map what they want, when they want. This openness is key to putting as few barriers as possible between mallets and the map.

OpenStreetMap has been key in places such as Haiti post-earthquake where it was used as the defacto map base between government agencies, NGOs and rescue teams in place of clumsy closed and overlapping maps. This has been claimed to save lives.

K. Tanaka, P. Fröhlich, and K.-S. Kim (Eds.): W2GIS 2011, LNCS 6574, p. 4, 2011.

GeoST: Geographic, Thematic and Temporal Information Retrieval from Heterogeneous Web Data Sources

Felix Mata[1] and Christophe Claramunt[2]

[1] UPIITA – IPN Avenida Instituto Politécnico Nacional No. 2580, Colonia Barrio la Laguna Ticomán Delegación Gustavo A. Madero, CP. 07340
mmatar@ipn.mx
[2] Naval Academy Research Institute, Brest, France
christophe.claramunt@ecole-navale.fr

Abstract. Geographical data commonly available on the Web is often related to thematic and temporal data, and represented in heterogeneous data sources. This makes geographical information retrieval tasks non straightforward processes as the information available is often unstructured. The research presented in this paper introduces an ontology-driven approach whose objective is to facilitate retrieval of geographical information on the Web. Spatio- temporal queries are categorized in order to solve questions related to several complementary modeling dimensions: *when*, *what* and *where*. The retrieval strategy is based on the exploration of a spatio-temporal *ontology*, and implemented by a search engine. The whole approach is exemplified by a prototype development, and the results obtained are evaluated by a metrics based on recall and precision measures. The experimental findings show that the strategy is effective for the types of spatio-temporal queries used by our geographical retrieval approach.

Keywords: Geographic and temporal information retrieval, Ontology, Web information retrieval.

1 Introduction

Nowadays, geographic, temporal and thematic information associated to a given phenomenon is usually distributed over numerous heterogeneous repositories such as the Web, but with not always explicit association or integration. Moreover, geographical information when available on the Web is often unstructured and represented using different data semantics and structures. One common problem users interacting with geographical information on the Web face is the correct expression of search queries in an appropriate form. For example: how to combine an event with entities and locations? How to define and approximate a query whose semantics is not completely well defined such as “Where is the Latin American tower? What places are near this tower? Why is this monument named so? How to go there?” Obviously such queries make sense for a Mexican, but not for others. It is also imprecise regarding the level of abstraction required in space, the time frame to search for etc.

K. Tanaka, P. Fröhlich, and K.-S. Kim (Eds.): W2GIS 2011, LNCS 6574, pp. 5–20, 2011.

The problem outlined by such query examples underlines the need for a methodology and modeling approach that handles in an integrated way the spatial, temporal and thematic semantics of some documents available on the Web, and the expression of these queries. Moreover, such an approach should also take into account the large variety of data sources available on the Web, in order to solve the problem adequately.

The research presented in this paper is grounded on recent progress in the modeling of geographical and temporal semantics, and on Geographic Information Retrieval (GIR) principles. Retrieval of geographical information has been long considered by geographical information systems. However, when considering the Web, retrieval of geographical information is a problem of different complexity as the information available is diverse and unstructured per nature. Contextual, temporal and thematic properties, associations between different geographical data sources and documents imply the development of novel retrieval mechanisms. Usually, retrieval approaches are centered on just one search criterion, for example using geographical and topological operations [24,26]. In contrast, the approach presented in this paper considers the whole range of geographical, temporal and thematic dimensions. The methodology developed extends a previous approach oriented to geographical information retrieval [20], hereafter extended to the temporal and thematic domains. The issues addressed and the expected contributions are threefold: 1) An ontology-based approach for the retrieval of geographic information, where geographical entities are qualified by temporal, spatial, and thematic relations among them; 2) An extension of our previous framework [20] to process queries taking into account geographical, temporal and thematic parameters in order to retrieve geographical information on the Web; 3) A context-oriented query definition engine that includes a large variety of spatial, thematic and temporal parameters applied to the search of some geographical entities.

The remainder of the paper is organised as follows. Section 2 briefly surveys related work. Section 3 introduces the principles behind the ontology design and the data modelling approach. Section 4 develops the retrieval strategy in the temporal, thematic and geographical domains, and the experimental results. Section 5 presents the evaluation of the results obtained. Finally section 6 draws the conclusions as well as future work.

2 Related Work

Time, thematic and space have been widely explored in the GIS literature, and this in several directions either at the conceptual, logical, or implementation levels [24,26]. Still, the way time, theme and space are associated to geographical entities are often represented in isolated and separated ways. Consequently, information processes oriented to the retrieval, management and analysis of multi-dimensional data are challenging but promising tasks when one wants to perform a search query that matches these different domains [5]. This entails the need for a domain ontology that closely integrates the time, thematic and geographical dimensions within an homogeneous information reference.

The potential of ontologies for the analysis and manipulation of geographical data has been already suggested [1]. It appears that ontologies and semantic annotations are likely to contribute to the development of more efficient retrieval processes [14, 15]. An example of such approach is described in [3], where an integrated spatio-temporal ontology has been developed in order to capture interactions between these different dimensions; the approach has been applied to several data sources. These semantics can be processed using first order logic and Web-based ontology languages such as OWL (Web Ontology Language). Other GIR recent efforts are oriented to the generation of a domain ontology based on several gazetteers and data resources such as the WorldNet general domain ontology and Wikipedia [6]. A different approach has been developed in a related work, instead of using a gazetteer; common sense geographic knowledge and qualitative spatial reasoning have been applied for the extraction of a geographic ontology [27]. Geographic search queries have been extended using a domain ontology represented as a semantic view of the data stored in a geographical database [7]. Integration of the geographical and thematic dimensions has been also considered in information retrieval [17,19]. A categorization of GIS tasks and GIR queries has been suggested in [19], the approach being applied to heterogeneous sources, including multimedia sources. A spatially-aware search engine (SPIRIT) oriented to the manipulation of geographical information has been developed [17]. The design and framework is based on a geo-ontology and on the integration of textual and spatial indexing of web documents.

When combined with query languages, domain ontologies favor the design and development of domain-based search engines and their application to different areas such as in transportation [16]. This often implies to develop specialized Information Retrieval (IR) mechanisms such as the Vector Space Model [4], which is based on the application of lexicographic term matching between queries and documents. When combining GIR and GIS systems, matching processes are based on the application of rule-based and data-driven methods that allow access to data resources based on spatial queries. One common trend of these ontological approaches is that they have been mainly oriented to the spatial and thematic semantics without a close integration of the temporal domain.

A spatio-temporal integrated ontology has been recently suggested, but geographical information retrieval is not the main goal, and the work is still preliminary [28]. In a recent and related work, time, space and thematic concepts have been integrated for the generation of a multi-language search engine for the manipulation of history textbooks [25]. The interface developed offers several search capabilities based on metadata extracted from Wikipedia documents. Queries are mostly keyword based and predefined search criteria. Our approach also considers the development of a geographic, thematic and temporal information retrieval engine applied to a set of reference documents such as Wikipedia and as in [25], but our framework differs in the sense of the search engine developed is based on whatever form of queries as far as it integrates the conceptual, thematic, and temporal domains, and not on predefined metadata. The search queries give what relevant data should be retrieved from a particular data source (i.e., Wikipedia in our case), and by using some criteria in the spatial, thematic and temporal domains. The approach is based on the belief that ontology, support of the approach, can favor retrieval of geographic information on top of these heterogeneous data sources on the Web.

3 Ontology Approach and Data Modeling

The research presented is based on a domain ontology that integrates the geographic, temporal and thematic dimensions related to some geographical objects of interest. The data sources used in this work are Wikipedia web pages that present a frame of reference for the integration of heterogeneous data sources (the Spanish version of Wikipedia is the one used for our work at http://es.wikipedia.org). The information contained in the geographical database used as an illustration of our approach is provided by the *National Institute of Statistics*, *Geography and Informatics* of Mexico (INEGI) and the *Secretary of Communications and Transportation* of Mexico (SCT).

3.1 Ontology Structure

A reference ontology is the core of our approach, it is built semi-automatically based on the principles introduced in [22], and our previous work that developed an integrated information retrieval engine [20]. Overall the ontology developed contains around sixty concepts and five levels of depth, the main structure of the ontology is illustrated in Figure 1.

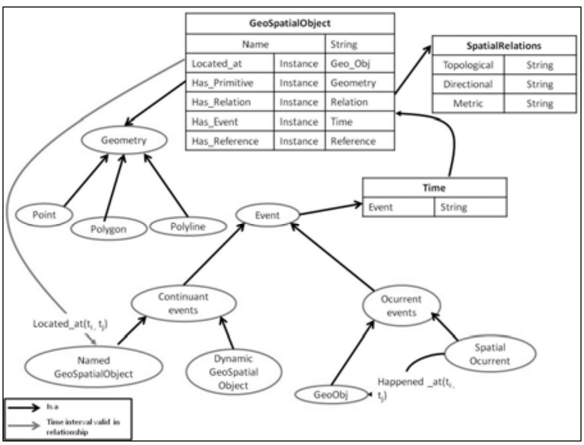


Fig. 1. Ontology Structure

In fact a reference ontology should contain and reflect the knowledge several users have on a given thematic, and the properties that make an entity member to such or such category should not be neglected (e.g., *Latin American tower, emblematic buildings and post-Mexican revolution domains*). Our reference ontology was built using thematic and concepts related to the country of Mexico and the Mexican Republican. A concept represents a geographical entity or an event of interest. Concepts are qualified by attributes, and associated by semantic, spatial, temporal and temporal relations. For instance, semantic relations can relate events and places, or concepts using hierarchical relationships (e.g., *part-of*, *is-a*). Geographical properties include metric and topological relations, while temporal properties are described by dates and the metric of time, and on an interval algebra [2]. Web pages are associated to concepts, as such they are termed as instances (e.g., a Web page about the Latin American tower is associated to a building concept). Finally, a context is defined as a set of neighboring terms (e.g., parent and sibling concepts) of a particular concept.

The format of the *XML* data structure is shown for the Latin American Tower modeled as a geographical entity and identified by a label (e.g., id 16034). The XML file references the documents associated to a given concept, that is, the Wikipedia documents where the Latin American Tower has been identified as part of the information provided (e.g., id 1789 and id 1567). The XML file also contains a description of some of the properties available in those documents (e.g., area, floor count), spatial relations (e.g., two towers located in the neighborhood of the Latin American Tower), some temporal facts (e.g., duration of the construction), and a few thematic properties (e.g., architect name). Overall the XML data structure favors the integration of the information identified for a given concept, and the relationships identified in the different domains. This allows for an extraction of relevant information from the geographical database and other Web documents.

```
<XML>
<Concept id="16034">
  <Documents>
     <id1>1789</id1>
     <id2>1567</id2>
  </Documents>
  <properties>
   <area > 28000 m2</area>
   <Floor count> 44 </Floor count >
 </properties>
  <spatial_relations>
      <near1>Caballito Tower</near1>
      <near2>Tlatelolco Tower</near2>
  </spatial_relations>
   <location>
        <lat>19°26′02″N_</lat>
        <lng>99°08′26″W </lng>
 </location>
 <temporal_relations>
    <rel1>Built</rel1>
   <event1_starts>1956</event1_starts>
   <event1_finishes>1972<event1_finishes>
 </temporal_relations>
 <time_reference>
   <name1>50 anniversary</name1>
   <year1>2006</year1>
</time_reference>
<thematic_relations>
   <name1>developer_La Latinoamericana_Seguros de Vida</name1>
    <name2>architect_ Augusto H. Alvarez</name2>
</thematic_relations>
 </Concept>
 </XML>
```

The ontology developed provides a basis for the development of a retrieval engine. It indicates what information should be retrieved taking into account the semantics of some temporal, thematic and geographical criteria. For example, let us consider the query Q_1 = {"When and where happened the Mexican Revolution?"}. First, one need to know what Mexican Revolution is, and which geographical entities and events are associated to it. This is achieved by exploring the reference ontology in order to find a concept that matches with Mexican revolution. If that concept is found, then the context is extracted and each dimension of the context (time, thematic and space) represents the search parameter to apply to the document sources.

The query approach uses a triplet format {what, where, relation}, where the term "what" denotes a geographical entity, the element "where" a geographic location or a geographical entity. Finally, the term "relation" denotes a semantic association between the "where" and "what" dimensions. The range of queries potentially applicable is illustrated by some examples presented in Table 1.

Table 1. GeoSt Queries

Query type	Query structure	Query Examples
Spatial query	*{Where, Event/Object}*	Q_2= {Where is the Upiita campus?}
Spatio-Temporal Query	*{Where & When, Event/Object}*	Q_3= {Where and When occurred the epidemic AH1N1?}
Spatial query	*{Where, Spatial Relation,Event/Object}*	Q_4= {Which are the hotels near airport Benito Juarez?}
Temporal query	*{When, Event/Object}*	Q_5= {When was built the Latin American tower?}
Spatio-Temporal	*{When, Spatial Relation,Event/Object}*	Q_6= {How much time to go from Mexico City to Cancun?}
Spatio-thematic Query	*{What, Spatial Relation,Event/Object}*	Q_7= {Which earthquakes have happened in Mexico City?}

Table 1 shows some query examples that combine spatial, temporal, and thematic criteria. Conjunction operators are applied when necessary. The temporal, theme and spatial domains are used either as criteria or as the data returned as a result of the queries. Events and objects are the main primitives manipulated (e.g., epidemic AH1N1 in query Q_3, Mexico city in Query Q_7, respectively). Relations are also integrated in these queries and this for the different domains (e.g., near as a qualitative spatial operator in query Q_4).

4 Retrieval Methodology

Overall, the retrieval methodology is ontology driven. The retrieval strategy is based on the search for a matching between query terms and corresponding concepts in the ontology. Concepts represent either events that happen or geographical entities [12]. A distinction is made between continuants and occurrents as suggested in [13]. Continuants denote entities which have continuous existence in time, and preserve their identity through change (e.g., a person, a city). Occurrents represent events and processes; they happen and then no longer exist (e.g., an epidemics, a displacement). The retrieval process starts by submitting a query into the system. The ontology is then explored in order to find a matching between the terms of the query and the concepts stored into that ontology. Each element of a query is searched within the ontology (e.g., by label names as each concept has a label name). When a concept is found into the ontology, then its context is captured. A context is formed by the neighboring concepts of a matched concept and its semantic relations are extracted and stored into a vector. Each search element is stored into a vector and represents a parameter applied as a query to the heterogeneous Web data sources. The process works according to the following algorithm depicted in pseudo code:

Begin

Initialization

1. Parsing and identification of each element of the query

Ontology exploration

2. Search (by concept name) the concept corresponding to the elements in the query
3. If (match)
4. Context extraction using the semantic relations of the ontology
5. Extraction of relations and properties
6. Set context, relations and properties into the context vector

Geographic Retrieval

7. Define the geographical search criteria data using the geographic parameters contained into the context vector

Thematic Retrieval

8. Define the thematic search criteria using the thematic parameters contained into the context vector relation

Temporal Retrieval

9. Define the temporal search criteria using the temporal parameters contained into context vector

Query execution

10. Execution of the query to the Geographical database
11. Execution of the Web service to the repository of Wikipedia documents
12. Execution of the Web service to the gazetteer *Geonames*

Query result storage

13. Integration of the query result into an XML file

End

The above algorithm illustrates how the different search parameters are used to find relevant information from the Web data sources (e.g., Wikipedia). A specific Web service *Geonames* has been developed in order to match a concept name to the terms of a gazetteer, and the matching is used as a parameter for the Web Service that is connected to the *Geonames* repository.

For example, let us consider the spatio-temporal query Q_1= {"*When and where happened the Mexican Revolution?*"}. The concept of the Mexican revolution is searched for into the ontology, when the concept is found, its context is extracted. The temporal context is a period of time [1910-1917], while the thematic context is related to a *social revolution*, and the geographical context is referenced to Mexico City and Northern Mexico. Figure 2 presents a graphic representation of this domain ontology, and part of the extracted context for query Q_1.

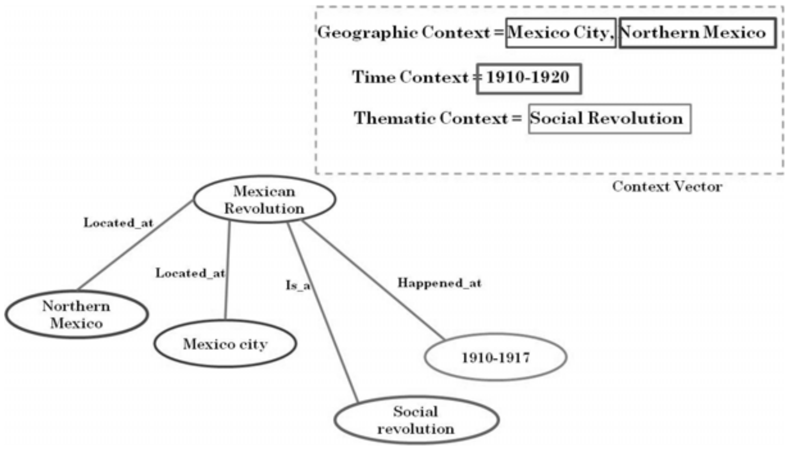


Fig. 2. Ontology and context for query Q_1

In essence, this search technique is similar to the one of the query expansion used in [9], but temporal attributes are not considered, while in [11] natural language expressions are applied. In contrast, matching is here performed according to the concepts and topological relations between the geographical objects considered by the query. Furthermore, the matching identified in some documents are associated to the concepts. For instance, in Figure 2 the concept "Mexican revolution" is related to the concepts "Mexico city" and "Northern Mexico" by the semantic relation "located at" and a period of time (1910-1917) by the temporal relation "event happened". This means that the context of concept "Mexican revolution" is associated to {"located at", "Mexico city", "northern Mexico", "event happened at", "1910-1917"}. This approach is extended to instances (i.e., documents) associated to these concepts. These documents are associated to each pair of concepts according to their semantic relations. The association criterion is a preliminary step of the retrieval process, it works as follows: queries on a given concept (e.g., Mexican Revolution) are submitted to Google and Yahoo Answers! Then, relevant results (i.e., web documents) are selected semi-automatically and associated to each concept into the domain ontology. This process is made using a XML file, which is structured to contain concepts, properties; spatial relations, temporal and thematic relations on the relevant geographical objects according to the ontology introduced in Figure 1. XML nodes are used to classify each one of them, and a node that references the documents associated to a given concept. A concept can represent some geographical entities related to temporal and thematic data. An XML file is used as a reference to ease the extraction of information from the Web pages and the geographical database. The thematic and temporal data integrated into XML files is extracted from the Web using a parser that scans online web pages to find the terms related to concepts. Place names and landmarks identified as terms are associated to geographical locations, and eventually related to some spatial and thematic data identified by the parser.

4.1 Retrieval Mechanisms: Principles

The retrieval mechanism is extended to some spatial and topological matching (e.g., two geographical objects related by a topological relation). When considering the query Q_1= {"*When and where happened the Mexican Revolution?*"} part of the obtained context is given by the terms {*"located at", "Mexico city", "event happened at", "Northern Mexico", is a Social revolution*}. The emerging context can be considered as a set of neighboring concepts; it is extracted from the hierarchical structure of the reference ontology. These concepts are related as subclasses, superclasses and sibling classes of the Mexican Revolution in the theme, temporal and spatial domains. For example, the Mexican revolution is associated to geographical concepts such as Mexico City and Northern Mexico, by the application of the spatial relation "located at". Next, a matching between these different concepts is searched for. When considering geographical entities the matching searched for is qualified by spatial associations such as topological relations. The data source used to find a match is the geographical database support of the approach. When a matching occurs, the data found is integrated to a final set of results to be displayed. For example, the match found for the context of query Q_1 is: Mexico City connected to Queretaro by highway 57, where each object (Mexico City, Queretaro, and highway 57) is identified, and associated with a spatial connection relation.

4.2 Retrieval from the Web

The Web information retrieval is performed on unstructured and freely available online data sources. A set of documents from the Spanish version of Wikipedia is collected and is considered as the Web information repository, as it contains a representative and large information space where all the information domains considered in our approach are available. This has also the advantage of reducing and bounding the search space as applied in other search mechanisms in the Web such as the Kleinberg algorithm [18]. Let us consider the query example Q_1= {"When and where happened Mexican Revolution?"}. The information is collected from a set of documents from the Spanish version of Wikipedia as available in January 2009. One of these documents is taken from http://es.wikipedia.org/wiki/Revolución_mexicana as this document is a relevant instance of concept Mexican Revolution. The search process is automated with a syntactical-semantic parser developed using the Ruby language [www.ruby-lang.org] In essence, the program browses the Web pages to find a matching with the context of the query considered. Regarding Q_1, the main concept searched is the "Mexican Revolution". Wikipedia pages have a predefined structure with a list of contents and paragraphs which are used as a starting point to find relevant information on the elements contained into the context. These paragraphs are scanned to find a term or terms of a given context. When a term is found, then the paragraph that contains the term is extracted. Next the sentences that contain the terms are extracted from the paragraph. Additional information that emerges from these sentences is searched for, that is, locations and place names, events and time references, and thematic information, the rest is discarded. Furthermore, when a match for a given date is found (e.g., a year from 1910-1917 period), the sentence that contains this date is also extracted. These sentences are syntactic and semantically processed, where stop words (e.g. *the, a, with,* etc.) are eliminated, and according to the semantic relations identified in the ontology, the relevant terms are extracted, and the sentences are integrated into a final set of results. Overall, the search process closely considers the context of the query in the following steps: the former searches for geographic locations and place names. The second one searches for dates, events, and time references. Finally the third one searches for additional thematic information using a Web service.

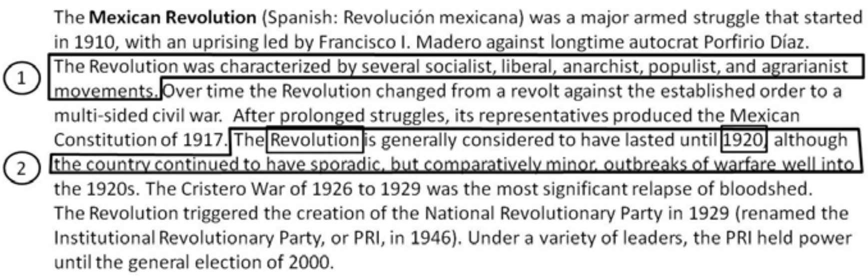

The **Mexican Revolution** (Spanish: Revolución mexicana) was a major armed struggle that started in 1910, with an uprising led by Francisco I. Madero against longtime autocrat Porfirio Díaz. (1) The Revolution was characterized by several socialist, liberal, anarchist, populist, and agrarianist movements. Over time the Revolution changed from a revolt against the established order to a multi-sided civil war. After prolonged struggles, its representatives produced the Mexican Constitution of 1917. (2) The Revolution is generally considered to have lasted until 1920, although the country continued to have sporadic, but comparatively minor, outbreaks of warfare well into the 1920s. The Cristero War of 1926 to 1929 was the most significant relapse of bloodshed. The Revolution triggered the creation of the National Revolutionary Party in 1929 (renamed the Institutional Revolutionary Party, or PRI, in 1946). Under a variety of leaders, the PRI held power until the general election of 2000.

Fig. 3. Ontology and context for query Q_1

Let us illustrate this search process using the query example Q_1. Figure 3 illustrates a fragment of Wikipedia page and how the sentences that contain a match for query Q_1 (e.g., Revolution) are labelled. When a match for a given date is found (e.g., a year from 1910-1917 period), the sentence that contains this date is also extracted. These sentences are syntactic and semantically processed, according to three steps: The first step starts searching a lexicographic that matches Q_1 and its context on top of Wikipedia pages. Several states and places related to the Northern of Mexico (given as the context) are retrieved (e.g., San Luis Potosi state, Puebla, Mexico city and other states of Mexican republic). These states and places are used as input parameters of a web service connected to the geographical database. The web service returns the geographical coordinates of these place names and states. The second step searches for some dates and temporal terms (e.g., start, finish, before, after) according to the query and the semantics exhibited by the web documents, and conceptualized by the ontology. In particular, the domain ontology identifies if the event qualifies an *occurrent* or *continuant*. For instance, for Q_1, we find that the Mexican revolution started in the year 1910 and finished in 1917, it is thus considered as an occurrent. The third and final step consists of finding additional thematic information according to the query and its web context. This process is clearly derived from Wikipedia-based information. For example, the Mexican revolution is described as a set of socialist, liberal, anarchist, populist, and agrarians movements, additional information can be similarly derived for the different places and states identified. The query results derived by the application of this search process are inserted into a XML database whose format is defined to contain all the results according to the temporal, thematic and spatial domains. A Web map is finally derived using a geographical server where each geographical place related to the Mexican revolution is identified and labeled (Figure 4).

Time Reference	Event starts	Event finishes	Geographic places related	Identifier
1910			SLP, San Luis Potosí City	C
1910	-	-	Mexicali City	A
1910	-	-	chihuahua City	B
-	1910	1917	Queretaro City	D
1917	-	-	Puebla City	F
-	1910	1917	Mexico City	E

Fig. 4. Query Q_1 results

As illustrated in Figure 4, a map displays the geographical locations related to Q_1, and below, the places related to the period of time (i.e., when an event starts and finishes). In addition, a reference time is given when appropriate. Furthermore, the approach uses Google maps API to display the search results according to a temporal timeline (i.e., depicted by an oriented line that connects the locations), that is, from the ordered sequence of dates related to the Mexican cities involved in the revolution (Figure 5).

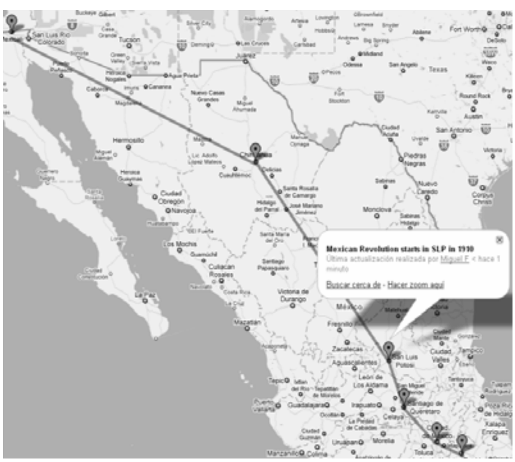


Fig. 5. Time-ordered results for query Q_1

The example developed illustrates the potential of the approach regarding the extraction of knowledge on the spatial, thematic and temporal domains. A large range of searches can be applied under similar principles. Figure 6 shows the results returned by the execution of query Q5= {When was built the Latin American tower?}.

Time Reference	Event starts	Event finishes	Geographic places related	Identifier
1956			Madero	A
1972	-	-	Eje central Lázaro Cárdenas	B
	1948	1956		
1987			Centro Histórico de la Ciudad de México.	C

Fig. 6. Query Q_5 results

Figure 6 displays the geographical locations related to query Q_5, and below, the places and time references related to this tower. In that case, the streets where the Latinamerican Tower is located are displayed (e.g., Madero, Lázaro Cárdenas), together with an event related to its construction and a reference to the historic center of the city of Mexico as the Latinamerican Tower also belongs to this historic concept.

5 Retrieval Evaluation

The effectiveness of our approach has been evaluated with the metrics of precision and recall, which are commonly used for testing IR and GIR strategies [8]. Precision is defined as the ratio of the number of relevant documents retrieved (*A*) to the total number of irrelevant (*B*) and relevant documents retrieved (*A*). Recall is defined as the ratio of the number of relevant documents retrieved (*A*) to the total number of relevant documents retrieved (A) and not retrieved (C). Precision and Recall are respectively given by

$$Precision = A / (A+B)$$
$$Recall = A / (A+C)$$

Recall and precision are typically valued by the unit interval [0, 1]. Amongst the set of documents considered by our experimental setup, we ask a group of twenty five students to value the documents that are considered as relevant or irrelevant for the example queries. Also the number of relevant documents for each query in the

Wikipedia search space has been approximated. Execution of a search query allows us to identify how many relevant documents were retrieved or not by the search strategy, as well as for irrelevant documents. Table 2 presents the statistical results obtained with the selected queries, and the values obtained for recall and precision.

Table 2. Recall and precision

Query	Irrelevant documents retrieved	Relevant documents retrieved	Relevant documents not retrieved	Total documents retrieved	Recall	Precision
Q_1= {"When and Where happened Mexican Revolution?"}	18	80	15	98	0.84	0.81
Q_2= {Where is the Upiita campus?}	35	70	14	100	0.83	0.66
Q_3= {Where and When occurred the epidemic AH1N1?}	22	85	17	110	0.83	0.79
Q_4= {Which are the hotels near airport Benito Juarez?}	31	70	14	110	0.84	0.69
Q_5= {When was built the Latin American tower?}	34	76	12	110	0.84	0.69
Q_6= {How much time to go from Mexico City to Cancun?}	26	60	16	86	0.84	0.69
Q_7= {Which earthquakes have happened in Mexico City?}	28	74	18	102	0.80	0.72

The results obtained reveal different patterns. Overall, the recall values obtained are relatively similar, those showing a relative high ratio of relevant documents retrieved, around one out of five documents retrieved being irrelevant (note that amongst the set of documents evaluated, the ones considered as relevant or irrelevant are know *a priori*).

More interesting, the precision values obtained show that a combination of spatial and temporal criteria are likely to increase the relevance of the documents retrieved as the combination of these two dimensions narrows the search space when those are expressed without too much ambiguity (e.g., Q_1 and Q_3). Conversely, queries that consider either the temporal or spatial dimension vary in terms of precision: the ones that perform not very well are the ones with either a qualitative spatial relation, those being more difficult to evaluate without ambiguity (Q_4), or a search for a temporal metrics as a query result (Q_6). Also there are cases where the search query is relatively precise from a linguistic point of view, this being the case of the temporal query Q_5. The precision is here relatively good because the context involves a search parameter that reduces the possibility of ambiguity, as combination of a keyword "built" and a given place "Mexico City" narrows the search space.

6 Conclusion

The research presented in this paper introduces a modelling and search approach to perform geographic information retrieval associated to thematic, temporal and geographical entities, events extracted from the Web and a geographical database. Retrieval processes are driven by a domain ontology. The queries used are contextualized in order to closely relate geographical information to thematic and temporal events derived from Web resources. The approach is based on a method that matches user-defined queries with ontological concepts according to the temporal, thematic and geographical domains. Retrieval is executed on top of internet resources and geographical databases. Place names, time information, as well as spatial relations are used as query parameters by the search engine. The engine developed and implemented so far combines XML-based queries with a parser and Google Map API. XML allows a close integration of the query results and support integration of information from heterogeneous data sources. The approach has been experimented on an illustrative case study applied in several regions of Mexico.

Further work will consider larger Web information spaces and document collections, and an integration of several distributed geographical databases as well as additional spatial relations. The domain ontology can be also enriched by considering additional spatial relations, and by application-based conceptualizations. The approach can be also applied to other ontologies available on the Web and free online resources and collections to compare with other IR approaches.

Acknowledgments

The authors of this paper wish to thank the Mobile Computing Laboratory-UPIITA, Instituto Politécnico Nacional in Mexico, IPN-UPIITA, COFAA-IPN, and SIP for their support, and the anonymous reviewers for their constructive comments and suggestions.

References

1. Agarwal, P.: Ontological considerations in GIScience. International Journal of Geographical Information Science 19(5), 501–536 (2005)
2. Allen, J.F.: Maintaining knowledge about temporal intervals. Communications of the ACM 26(11), 832–834 (1983)
3. Arpinar, I.B., Sheth, A., Ramakrishnan, C., Usery, L., Azami, M., Kwan, M.: Geospatial ontology development and semantic analytics. Transactions in GIS 10(4), 551–576 (2006)
4. Baeza-Yates, Ribeiro-Neto, B.: Modern Information Retrieval. ACM Press, Addison-Wesley, New York (1999)

5. Baglioni, M., Masserotti, M.V., Renso, C., Spinsanti, L.: Building geospatial ontologies from geographical databases. In: Fonseca, F., Rodríguez, M.A., Levashkin, S. (eds.) GeoS 2007. LNCS, vol. 4853, pp. 195–209. Springer, Heidelberg (2007)
6. Buscaldi, D., Rosso, P., García, P.: Inferring geographical ontologies from multiple resources for geographical information retrieval. In: Proceedings of 3rd Int. SIGIR Workshop on Geographic Information Retrieval, SIGIR, Seattle, pp. 52–55 (2006)
7. Cardoso, N., Silva, M.J.: Query expansion through geographical feature types. In: GIR 2007: Proceedings of the 4th ACM Workshop on Geographical Information Retrieval, New York, NY, USA, pp. 55–60 (2007)
8. Clarke, S., Willett, P.: Estimating the recall performance of search engines. ASLIB Proceedings 49(7), 184–189 (1997)
9. Delboni, T.M., Borges, K.A., Laender, A.H., Davis, C.A.: Semantic expansion of geographic Web queries based on natural language positioning expressions. Transactions in GIS 11(3), 377–397 (2007)
10. Egenhofer, M.: Interaction with geographic information systems via spatial queries. International Journal of Visual Languages and Computing 1(4), 389–413 (1990)
11. Fu, G., Jones, C.B., Abdelmoty, A.I.: Ontology-based spatial query expansion in information retrieval. In: Liu, Y., Jiang, T.-Z., Zhang, C. (eds.) CVBIA 2005. LNCS, vol. 3765, pp. 1466–1482. Springer, Heidelberg (2005)
12. Galton, A.: On what goes on: The ontology of processes and events. In: Bennett, B., Fellbaum, C. (eds.) Proceeding of the 2006 Conference on Formal Ontology in Information Systems (FOIS 2006). Frontiers in Artificial Intelligence and Applications, vol. 150, pp. 4–11. IOS Press, Amsterdam (2006)
13. Grenon, P., Smith, B.: SNAP and SPAN: Towards dynamic spatial ontology. Spatial Cognition and Computation 4(1), 69–104 (2004)
14. Hammond, B., Sheth, A., Kochut, K.: Semantic enhancement engine: a modular document enhancement Platform for semantic applications over heterogeneous content. In: Kashyap, V., Shklar, L. (eds.) Real World Semantic Web Applications, pp. 29–49. IOS Press, Amsterdam (2002)
15. Harding, J.: Geo-ontology Concepts and Issues. In: Report of a Workshop on Geo-ontology, Ilkley, UK (2003)
16. Huang, B., Claramunt, C.: Spatiotemporal data model and query language for tracking land use change. Transportation Research Record, 107-113 (2005)
17. Jones, C.B., Abdelmoty, A.I., David Finch, D., Fu, G., Vaid, S.: The SPIRIT spatial search engine: architecture, ontologies and spatial indexing. In: Egenhofer, M.J., Freksa, C., Miller, H.J. (eds.) GIScience 2004. LNCS, vol. 3234, pp. 125–139. Springer, Heidelberg (2004)
18. Kleinberg, J.: Authoritative sources in an hyperlinked environment. Journal of the ACM 46(5), 604–632 (1999)
19. Larson, R.R.: Geographic Information Retrieval and Spatial Browsing, University of California, Berkeley, `https://sherlock.sims.berkeley.edu/geo_ir/PART1.html` (last access March 2010)
20. Mata, F.: Geographic information retrieval by topological, geographical, and conceptual matching. In: Fonseca, F., Rodríguez, M.A., Levashkin, S. (eds.) GeoS 2007. LNCS, vol. 4853, pp. 98–113. Springer, Heidelberg (2007)
21. Noy, N.F., McGuiness, D.L.: Ontology Development 101: A Guide to Creating Your First Ontology, Stanford Knowledge Systems Laboratory Tech. Rep. KSL-01-05 and Stanford Medical Informatics Tech. Rep. SMI-2001-0880 (March 2001)

22. Perry, M., Hakimpour, F., Sheth, A.: Analyzing theme, space, and time: an ontology-based approach. In: Proceedings of the 14th Annual ACM international Symposium on Advances in Geographic Information Systems, GIS 2006, pp. 147–154. ACM Press, New York (2006)
23. Petras, V., Fredric, C., Gey, L.R.: Domain-specific CLIR of English, German and Russian using fusion and subject metadata for query expansion. In: Peters, C., Gey, F.C., Gonzalo, J., Müller, H., Jones, G.J.F., Kluck, M., Magnini, B., de Rijke, M., Giampiccolo, D. (eds.) CLEF 2005. LNCS, vol. 4022, pp. 226–237. Springer, Heidelberg (2006)
24. Peuquet, D.J.: Representations of Space and Time. The Guilford Press (2002)
25. Pfoser, D., Efentakis, A., Hadzilacos, T., Karagiorgou, S., Vasiliou, G.: Providing universal access to history textbooks: a modified GIS case. In: Lytras, M.D., Damiani, E., Tennyson, R.D. (eds.) WSKS 2008. LNCS (LNAI), vol. 5288, pp. 87–102. Springer, Heidelberg (2008)
26. Worboys, M.: Event-oriented approaches to geographic phenomena. International Journal of Geographical Information Science 19, 1–28 (2005)
27. Yi, Z., Yong, G., Lu, X., Si, S., KaiChen, C.: A common sense geographic knowledge base for GIR. Science in China Series E: Technological Sciences 51(suppl. I), 26–37 (2008)
28. Zhaoqiang, H., Wenling, X., Xiuwan, C.: Spatial temporal geographic ontology. In: Geoscience and Remote Sensing Symposium International, pp. 4627–4630. IEEE, Los Alamitos (2007)

Extraction and Geographical Navigation of Important Historical Events in the Web

Mitsuo Yamamoto[1], Yuku Takahashi[2], Hirotoshi Iwasaki[1], Satoshi Oyama[3], Hiroaki Ohshima[2], and Katsumi Tanaka[2]

[1] Denso IT Laboratory, Shibuya CROSS TOWER 28th Floor 2-15-1, Shibuya, Shibuya-ku, Tokyo 150-0002, Japan
[2] Graduate School of Informatics, Kyoto University, Yoshida-Honmachi, Sakyo-ku, Kyoto, Kyoto 606-8501, Japan
[3] Graduate School of Information Science and Technology, Hokkaido University, Kita 14, Nishi 9, Kita-ku, Sapporo, Hokkaido 060-0814, Japan

miyamamoto@d-itlab.co.jp, ytakahas@dl.kuis.kyoto-u.ac.jp, hiwasaki@d-itlab.co.jp, oyama@ist.hokudai.ac.jp, ohshima@dl.kuis.kyoto-u.ac.jp, tanaka@dl.kuis.kyoto-u.ac.jp

Abstract. We propose techniques for achieving the geographical navigation of historical events described in Web pages as "Virtual History Tour". First, we develop a method for extracting information on the historical events from the Web and organizing it into a chronological table. Our method can effectively handle ambiguous cases – homonyms and multiple location names in a sentence – by using the number of co-occurrences among events, person names, location names, and addresses in the Web. Next, we propose a method for ranking historical entities according to their impacts at specific time and location. We extend the PageRank algorithm to calculate the temporal and spatial impacts of entities. Finally, we introduce our concrete application demonstrating how users can browse historical events through timeline and map interfaces.

1 Introduction

In recent years, there has been a huge amount of historical information placed on the Internet that can be accessed easily. Using this information, people can easily gather Web pages related to a certain historical person or happening.

We propose a method to aggregate such historical information in the Web and to generate a chronological table for a given historical person or happening. In particular, the proposed method tries to identify the location and the time of each event in the chronological table wherever possible. We also develop an application called "Virtual History Tour", which achieves the navigation of the extracted events on a map interface with automatic movements in chronological order. This is a novel means of providing information in which users can trace the life of a specified person or collect all the historical incidents that occurred at a specified location.

In order to achieve this service, we need a technique for extracting all the historical events from the Web and for determining their location and time by

K. Tanaka, P. Fröhlich, and K.-S. Kim (Eds.): W2GIS 2011, LNCS 6574, pp. 21–35, 2011.

inferring from their descriptions that is often written in an ambiguous and symbolic way or is often missing. In addition, we need a technique to add "impact" on the output extracted from the Web information.

In this paper, we explain the techniques we developed and their applications as follows.

– A technique to extract information on historical incidents relative to a specific incident from the text available in the Web, and to classify it by time and location: in particular a technique for data inference with the help of "Collective Intelligence".
– A technique to compute the "Impact" of historical entities.
– A demonstration to the application called "Virtual History Tour",that we developed, to evaluate how available information can be used with the above techniques.

2 Related Work

There are several environments for the visualization of a historical event sequence. Our system is implemented on Google Earth[1], which is an application that allows for the easy creation of geographical applications. Google Map[2], a map service on the Web, provides an API that can be easily used to create many kinds of geographical Web services. Stoev et al. [18] provides a visual environment especially for the historical events.

There are several related techniques and applications that are proposed in the research field of Geographic Information Retrieval (GIR), which is defined by Larson [9,11]. Strötgen et al. [19] proposed a method for extracting spacio-temporal information from a single document. They also developed a system to explore an extracted sequence of spacio-temporal information on a Google Map interface. While their target is a single document, our proposed method is to aggregate information from many documents available on the Web. Several other similar applications have been proposed by Larson's research group [12,10,5,17]. They focus on historical biography and visualize a sequence of life events in time and space. Their system places the historical events of a target person on a Google Map interface. The historical events related to a location are explained by clicking a place marker. The explanation usually contains temporal information, but it is difficult to find an explanation that describes the events in chronological order. Geocoding is a technique for identifying the latitude and longitude for a given geographic document. There are several researches regarding this technique [1,2]. We propose a method for extracting the location of an event by using an existing geocoding system.

Information extraction is another research area related to our work. In particular, Web information extraction has become a large research field. For example, Paşca proposed methods to extract factual information from a large Web corpus [16] and query logs [15]. Although most of the research studies focus

[1] Google Earth: http://earth.google.com/
[2] Google Maps: http://maps.google.com/

on *current* fact extraction, some focus on *historical* fact extraction. Garera et al. [4] proposed a method for biographic fact extraction such as "birthdate", "occupation", "nationality", and "religion". BioSnowball by Liu [13] is a method for gathering and integrating biographical facts. Their method can construct Wikipedia-stype pages for any person by aggregating the extracted facts. The values of the attributes of a person can change over time. When extracting the values from the Web, such changes can also be considered [14]. For temporal information extraction, not only the past events but also the future events can be extracted. Jatowt et al. [7] proposed a method for future event extraction from news articles or Web pages. They mainly focused on a method for extracting expressions that predict the future.

3 Historical Event Extraction

3.1 Extracting Historical Events from the Web

This section explains a method called "Historical Event Extraction" which extracts chronological events relevant to a specific person or happening, and classifies them according to time and location. In our previous work [8], we developed a method to gather chronological events about a specific person or happening and to estimate the time when each extracted event occurred. Adding to the previous work, we developed a method for estimating the location of an extracted event.

Fig. 1 shows the procedure for extracting historical events related to a specific person or happening. It consists of three steps as follows:

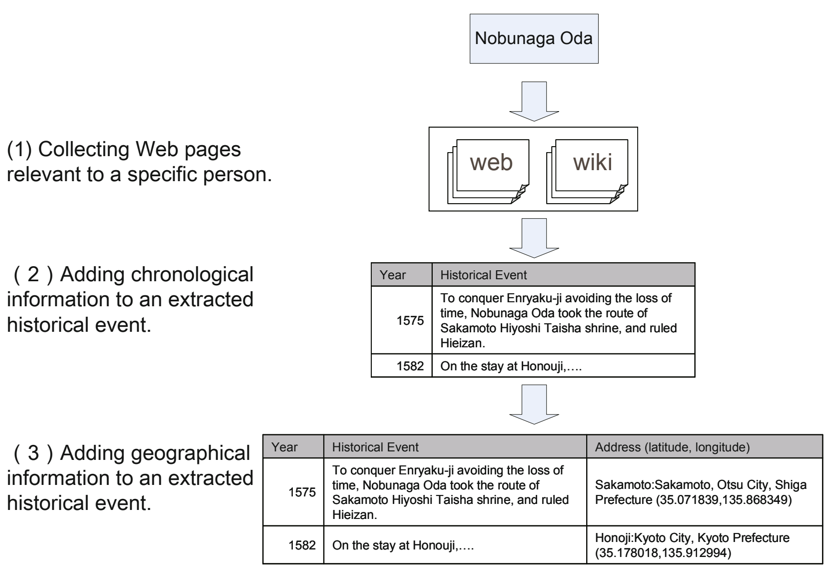


Fig. 1. Procedure for Chronological Event Extraction

1. Collecting Web pages relevant to a specific person.
   Let a user specify a word that represents the target person or happening, which is used as an inquiry keyword on a conventional Web search engine such as Google[3] and Yahoo![4]. Then collect as many Web pages that as possible that hit on the query.
2. Adding chronological information to an extracted historical event.
   From the Web sites collected in step 1, search words or expression that can be a clue to adding chronological information, and modify these words or expressions in a predefined way.
3. Adding geographical information to an extracted historical event.
   Extract the name of the event and its location by applying segmentation to the information acquired in step 2, and determine the longitude and latitude where the event occurred.

Steps 1 and 2 are basically the same as the methods described in the previous work [8] except for some enhancements such as the use of better regular expressions to extract temporal information. These methods can efficiently aggregate and normalize temporal data that is written in a different style.

The step 3, the location estimation of historical events, is one of the contributions of this paper. In many cases, it is possible to determine the specific locations of the extracted events by using a method called "Location Estimation of Event with Collective Intelligence" The rest of this section describes and evaluates this method.

3.2 Location Estimation of Historical Event

Although the name of a certain location can vary depending on the time or era, we were able to safaly determine its location from the description safely by using other resources such as historical materials. However, there still exist some problems that need to be solved as follows.

The context problem: Multiple different location names in a single sentence. It is often the case where there are more than just one location name appears on a single sentence. For example, think of sentence below. *"To conquer Enryaku-ji avoiding the loss of time, Nobunaga Oda took the route of Sakamoto Hiyoshi Taisha shrine, and ruled Hieizan."*[5] If you apply automated segmentation to this sentence and extracts all the words associated with the geographical word, you will get three location names: "loss", "Sakamoto", and "Hieizan". Here, "loss" is not location name, but "loss" and "Los" of *Los Angeles* is repsented in same way in Japanese, so it is extracted as location name. Obviously, "loss" was extracted by mistake, so we need to determine which of the remaining the candidates to choose.

[3] Google: http://www.google.com/

[4] Yahoo! Search: http://search.yahoo.com/

[5] *Nobunaga Oda* is one of the most famous feudal rulers in Japanese in the late 16th century. (http://en.wikipedia.org/wiki/Oda_Nobunaga)

The homonym problem: The same name that has different locations For example, if you search geocoding with the keyword "Sakamoto", you will get two hits such as (1) Sakamoto-chyou, Yasiro City and (2) Sakamoto Otsu City. If you are familiar with this event, you should correctly choose (1), but this is not always the case.

To overcome these problems, we propose a technique called "Location Estimation of Events with Collective Intelligence".

3.3 Location Estimation of Event with Collective Intelligence

In our estimation method, we focus on the name of the place and its address in the sentence. Here, we assume that the name of the place and its address appears within the same document in the Web that describes the historical person and event. The notation we use in this section is summarized as follows.

- T_i denotes one of the location names extracted from sentence that explain a historical event.
- A_j denotes a text expression of one of the addresses retrieved from geocoding with a location T_i.
- H denotes a set of people's names extracted from the sentence.
- E denotes a set of happenings' names extracted from the sentence.

The following formula is a scoring function that calculate relevance between a location name and an address based on our assumption.

$$\begin{aligned} Score(T_i, A_j) &= P(T_i|H,E) \cdot P(A_j|H,E) \\ &= \frac{N_{H,E}(T_i)}{N_{H,E}} \cdot \frac{N_{H,E}(A_j)}{N_{H,E}} \\ &= \frac{WebHitCount(T_i,H,E)}{WebHitCount(H,E)} \cdot \frac{WebHitCount(A_j,H,E)}{WebHitCount(H,E)} \end{aligned} \tag{1}$$

where $P(A_j|H,E)$ is the conditional probability of the address A_j among $N_{H,E}$ documents, and $P(T_i|H,E)$ is the conditional probability of the location T_i among $N_{H,E}$ documents. They can be calculated by $N_{H,E}$, $N_{H,E}(T_i)$, and $N_{H,E}(A_j)$. $N_{H,E}$ is defined as the number of the document that includes the name of the person, H, and the event. $N_{H,E}(T_i)$ is the number of the document among $N_{H,E}$ documents that includes the name of the location. $N_{H,E}(A_j)$ is the number of the document among $N_{H,E}$ documents that includes the name of the address. We can obtain $N_{H,E}$, $N_{H,E}(T_i)$, and $N_{H,E}(A_j)$ by using a conventional Web search engine such as Google and Yahoo!, which returns *hit count* with a search result. Our implementation uses Google AJAX Search API[6]. Each of the words in T_i, A_j, H, and E is added to the keyword query as *AND* condition.

For the event sentence we show in Section 3.2, T_i can be either "loss", "Sakamoto" or "Hieizan". When T_i is "loss", A_j can be "California, Los Angeles". H is "Nobunaga Oda", and E is empty.

[6] Google AJAX Search API: http://code.google.com/intl/en/apis/ajaxsearch/

Table 1. $P(T_i|H,E)$

T_i	$P(T_i\|H,E)$
loss	0.0192
Sakamoto	0.3728
Hieizan	0.8065

Table 2. $P(A_j|H,E)$

A_j	$P(A_j\|H,E)$
Shiga Prefecture	0.2753
Otsu City	0.2674
Sakamoto	0.3728
Kumamoto Prefecture	0.0412
Yashiro City	0.0015
California	0.0014
Los Angeles	0.0087

Table 3. Score

T_i \ A_j	Shiga Prefecture	Otsu City	Sakamoto	Kumamoto Prefecture	Yashiro City	California	Los Angeles
loss						0.0000	0.0002
Sakamoto	0.1026	0.0997	0.1389	0.0154	0.0001		
Hieizan	0.2220	0.2156	**0.3006**	0.0332	0.0012		

Our proposed method is unique in the way that we estimate the location of the event by multiplying a term $P(T_i|H,E)$ with $P(A_j|H,E)$. Because each probability term is acquired independently in terms of T_i and A_j, this covers the case where the only address A_j appears but not the location name T_i appears in the document, and vice versa. After multiplication, we calculated Score defined by Eq.(1) above for each possible candidate address A_j. We selected the address A_j with the highest *Score* as the most likely address where the event occurred,along with its latitude and longitude. We show some examples of scores computed by the proposed method in Fig.3. In this example, the highest score is observed for T =Hieizan and A =Sakamoto, yielding the fact that Sakamoto Otsu City is most strongly related in the given set.

3.4 Evaluation of Location Estimation

In order to see the validity of our proposed method, especially the method proposed in Section 3.3, we evaluated the precision of location estimation.(conducted on September, 30, 2009) in the following manner.

1. We picked up three well-known historical persons such as "Yoshitsune Miyamoto", "Nobunaga Oda", and "Ryoma Sakamoto", and collected 40 historical events associated with these three people by performing a Google search.Here, historical events are chosen under the restriction in which both the name of the person and name appear in.
2. We estimated the locations of these 40 events based on our proposed method.
3. After having these events plotted on a map with their estimated locations, we asked a single examinee to evaluate the result subjectively.

Table 4. Example of historical persons and events used in the experiment

Search keyword	Historican event name of location(bold)	Person H	Event name E
Yositsune Miyamoto	Yoritomo ordered samurais in **Kanto** to serve in **Kyoto** and for bided returning to **EastArea** on April,15h.	Yoritomo	serving at Kyoto
Ryoma Sakamoto	In 1866, on November, Sakamoto persuaded **Satsuma** Lord Saisuke 5th and **Choshu** Lord Heisuke Hirosawa on the agreement to establish a joint venture at **Shimonoseki**.	Heisuke Hirosawa	The establishment of a joint venture

Table 5. Results

	number of correct
Conventional method	17(42.5%)
Proposed method	25(62.5%)

The table below shows some of the events among the 40 events were used.

The evaluation result as shown in Table 5, appears to be 20% more precise than a conventional method. Table 6 show the results of the estimation of the location names of the event. As the data shows, we were still unable to estimate the 37.5% of location of events. We think this is because of the following reasons: failure of the sentence segmentation to extract the correct geographical words, and the domination of the major name of places over the remaining minor places, which can be mistakenly treated as a noise and discarded. This will lead to the conclusion that our method alone is not so suitable to deal with minor events where value N is small, and we are hoping to handle minor event case with other inference method to enhance its availability.

Table 6. Result of Estimation

Search keyword	Historican event, name of location(bold)	Estimated location
Yositsune Miyamoto	Yoritomo ordered samurais in **Kanto** to serve in **Kyoto** and for bided returning to **EastArea** on April	Kyoto Prefecture
Ryoma Sakamoto	In 1866, on November, Sakamoto persuaded **Satsuma** Lord Saisuke 5th and **Choshu** Lord Heisuke Hirosawa on the agreement to establish a joint venture at **Shimonoseki**.	Shimonoseki City

4 Computing "Impacts" of Historical Entities

The impact size of a certain historical entity changes with time. There are also geographical impacts, for example an event that influences only Japan can occur.

In this section, a historical entity means a historical happening, place, person, or item. We assume that a historical article on Wikipedia represent the related historical entity. In this section, we propose an approach that assesses the temporal and spatial impacts of historical entities by using the Wikipedia link structure.

4.1 Wikipedia Link Structure Analysis

Review of PageRank. The main idea behind PageRank [3] is that a Web page is important if several other important Web pages point to it. This means that if page u has a link to page v, then the link implicitly confers some importance to v. How should we represent the value of conferred importance? Let $Rank(p)$ represent the degree of importance of page p, and let N_p represent the out-degree of page p. We can simply assume that all links are equal, so the link (u, v) confers $Rank(u)/N_u$ units of importance from u to v. This simple idea leads to the following recursion. If N is the number of pages, assign all pages the initial value $1/N$. Let B_v represent the set of pages that points to v. In each iteration, the ranks are shifted as follows:

$$\forall v \; Rank_{i+1}(v) = \sum_{u \in B_v} \frac{Rank_i(u)}{N_u} \tag{2}$$

We continue the iterations until all $Rank(p)$ stabilizes to within some threshold, although the convergence of the recursion depends on the link structure. We add the damping factor, α, to the rank propagation to guarantee the convergence. We define a new recursion, in which we add the constant value $(1-\alpha)/N$ to all pages:

$$\forall v \; Rank_{i+1}(v) = \alpha \sum_{u \in B_v} \frac{Rank_i(u)}{N_u} + \frac{1-\alpha}{N} \tag{3}$$

This modification improves the quality of PageRank by introducing the damping factor α, as well as facilitates to guaranteeing the convergence to a certain value for all pages.

Biased PageRank. We can bias the PageRank computation to increase the effect of certain categories of pages by changing the added value $(1-\alpha)/N$ described in above. In particular, let D_j be the set of pages in a certain category c_j. We introduce a new damping factor d_{ij} for the page i:

$$d_{ij} = \begin{cases} 0 & i \notin D_j \\ \frac{1}{|D_j|} & i \in D_j \end{cases} \tag{4}$$

Then when computing the PageRank, we calculate as follows:

$$\forall v \; Rank_{i+1}(v) = \alpha \sum_{u \in B_v} \frac{Rank_i(u)}{N_u} + (1 - \alpha) d_{vj} \tag{5}$$

In this recursion, pages that exactly fit a category and pages that are closely related get high scores.

The biased PageRank algorithm is also used in Topic-Sensitive PageRank [6]. For biasing, the 16 top-level categories of the Open Directory Project (ODP)[7] are used. The set of URLs included in each category is used as D_j.

We create the temporal and spatial categories for Wikipedia historical entities and discuss how D_j for category c_j is determined in Sections 4.2 and 4.3.

4.2 Temporal Impact Calculation

As discussed in Section 4.1, the basic idea implemented by a graph-based ranking model that includes PageRank is that a link of a Web page to another one has a social meaning (i.e. "voting", "recommendation", etc.) In addition, if a graph domain is specified, we are able to think on the domain specific meanings of the link. In our study, each node on a graph is a historical entity given by a Wikipedia article, so the links between these articles are not only a part of the World Wide Web, but also a meaningful relation between the two historical entities. We consider the linked relation as an impact of the history.

Our main idea is that a historical entity has a large impact on certain period if several other entities occurred at the same time have links to it. In our approach, we treat a certain period as a category c_j and a set of historical entities occurred in the period as D_j in order to use biased PageRank algorithm. We believe that the given score has good correspondence to the historical impact of the entity in c_j.

4.3 Spatial Impact Calculation

Similar to the temporal categories created to calculate the temporal impacts of historical entities, we also create spatial categories to calculate the spatial impacts of them. For example, the geographic information of Wikipedia entries and the "history of somewhere" Wikipedia categories are particularly useful.

4.4 Experiments

As a dataset for our evaluation, we used 33,000 Japanese Wikipedia entries which all belong to subcategories of the "History of Japan" Wikipedia category. An element of the dataset represents a historical entity. There are 593,000 links among these pages.

We divide 0 to 2009 into 201 ranges at 10 intervals and name each ranges $c_0 \cdots c_{200}$, respectively. To be specific, the first range c_0 means 0 to 9, and the last one c_{200} is 2000 to 2009. D_j is a set of historical entities which occurred in c_j.

[7] Open Directory Project: http://www.dmoz.org/

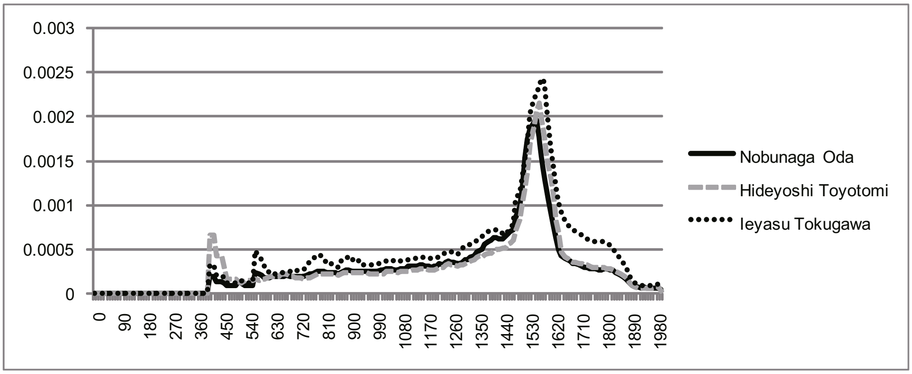


Fig. 2. Impacts of "Nobunaga Oda", "Hideyoshi Toyotomi", and "Ieyasu Tokugawa". The horizontal axis means A.D., not a category's index j. The vertical axis shows scale of biased PageRank.

For example, D_{194} includes "Pacific War" which broke out at 1941. We make a vector of 201 dimensions for all entities with these calculation.

The temporal impacts of the tree most important leaders: "Nobunaga Oda", "Hideyoshi Toyotomi", and "Ieyasu Tokugawa" in the Warring States period (16th century) are plotted in Fig.2. For all of them, the highest value is observed while they alive. During Edo period (1603-1868) "Tokugawa Ieyasu" has higher score than the other two. We think that this is the influence of his career; he established the Edo shogunate, which has great significance in Japanese history.

Fig.3 presents a comparison after the 19th century between the impacts of "Hiroshima Peace Memorial" and "Pacific War". The score of "Pacific War" reaches its peak around 1950; on the other hand the score of "Hiroshima Peace Memorial" is still increasing. We think that this is the influence of its designation as the UNESCO World Heritage Site in 1996.

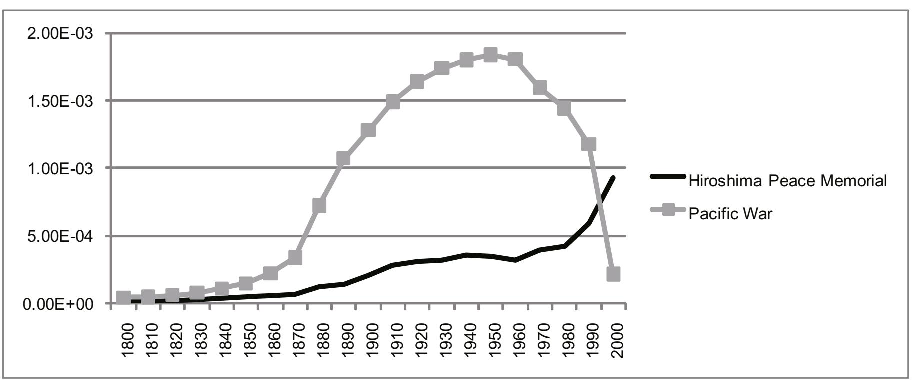


Fig. 3. Impacts of "Hiroshima Peace Memorial", and "Pacific War". Since these values are almost zero, the values of before the 19th century are not plotted in the figure.

5 Geographical Navigation of Historical Events

5.1 Application Overview

In this section, we explain the developed application, “Virtual History Tour”, which enables users to explore the historical events that are organized and sorted by chronological and geographical order, having been collected from Web sites. This application is features the following functions

1. The user can trace back relevant events in chronological order that are associated with a specified person or event.
2. The user can have historical events plotted on various types of maps ranging from an ancient map to a satellite photo and view them.
3. The user can browse photos associated with a specified event that are collected automatically from Web sites.

5.2 System Structure and Procedure

Virtual History Tour is composed of the following three parts:

1. History Time Table Generator
   By using the technique “Historical Event Extraction”, this generates a time table of events associated with a specified person and event is generated.
2. History Tour Planner
   This creates a tour plan along with the time table acquired by the Time Table Generator, together with photos collected from Web sites.
3. Browser
   This allows the user to browse the tour planned above.The current prototype works with Google Earth. Its procedure is shown in Fig.4.

5.3 Layout of Virtual History Tour

We show below, the result of tracing Ryouma Sakamoto[8]. Fig.5 shows the events associated with him that are plotted on the current map. Fig.6 shows the historical event details plotted on the old map. Fig.7 shows the superimposition of the Kyoto Edo-period old map on the current map.

5.4 Discussion

With our application, “Historical Event Extraction”, one can follow certain person’s life along time axis and his geographical range over his life, and learn to what geographical and chronological extend a single person or event has spread its influence.

[8] The demonstration video is avairable in the following URL.
http://www.dl.kuis.kyoto-u.ac.jp/~ohshima/historicalnavi/demo.wmv

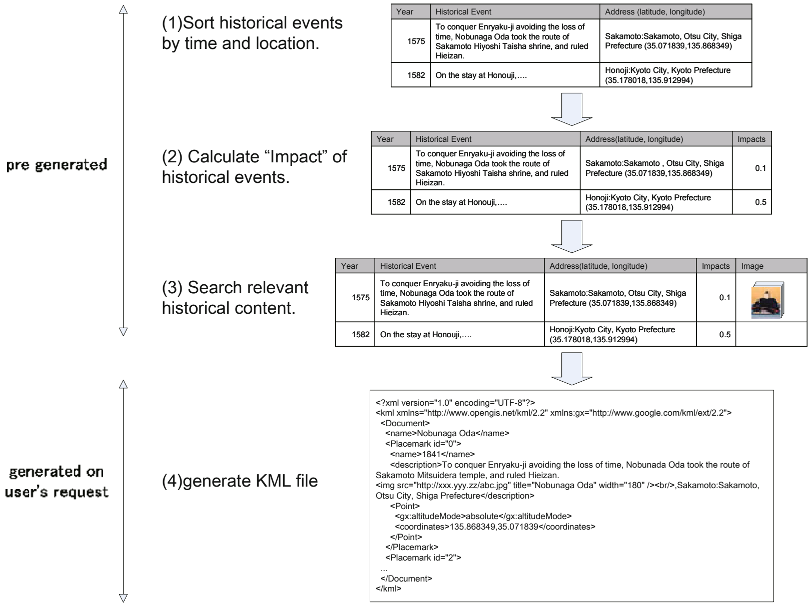


Fig. 4. Procedure for Historical Tour

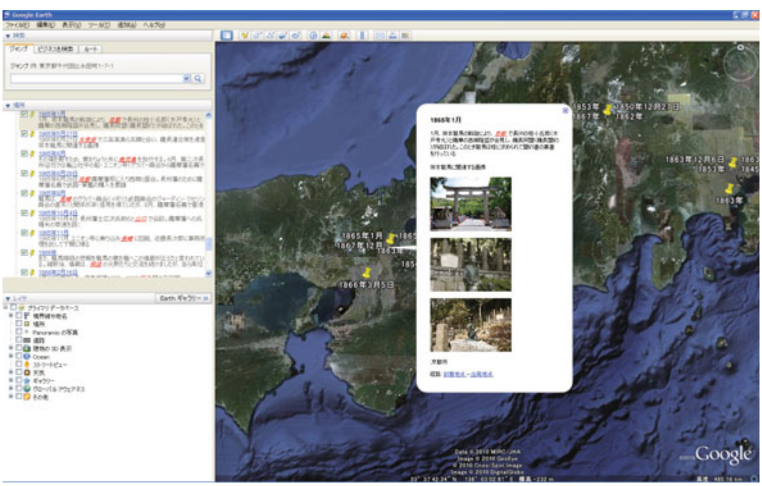


Fig. 5. Events associated with Ryoma Sakamoto plotted on the current map

At last, we consider the further possible use of our applications. First, it can offer information about recommended historical famous spots to stop by on traveling planning site. With this service, use can be aware of all historical spots around his destination, he can make more flexible plan of palaces to go.

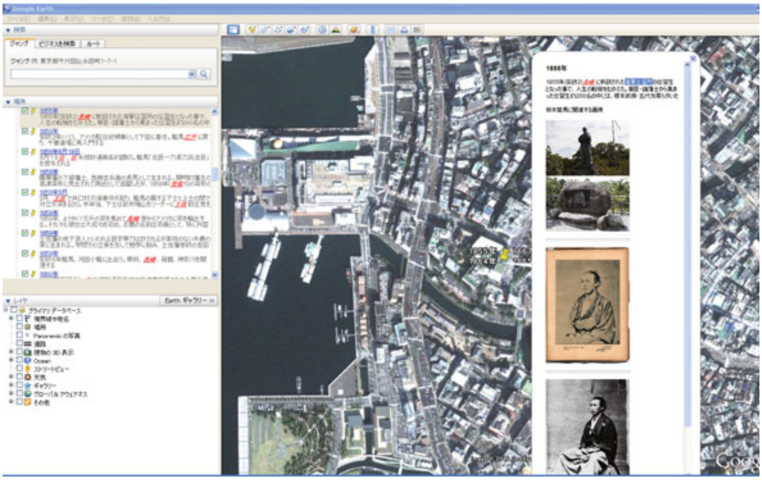


Fig. 6. Historical event detail plotted on the current map

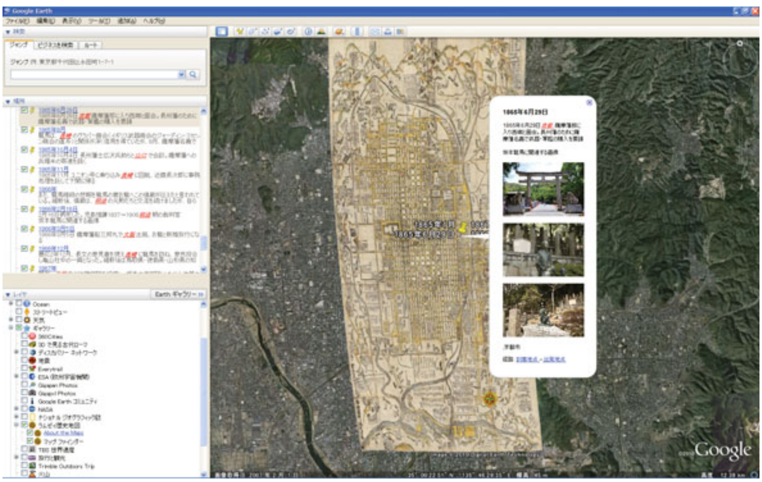


Fig. 7. Superimposition of Kyoto Edo-period old map to current map

This can be also applied to Car Navigation System, which offers drivers an unexpected history guide, by following a trait of historical figures, replacing the job done by bus guide concierge.

This guide feature can be further improved by adapting user's routine or preference, it lets a user decide what to learn next with his own, which helps user to learn history more pleasant way.

6 Conclusions

We proposed techniques for achieving the navigation of historical events described in Web pages. First, we developed a method for extracting chronological tables from the Web pages. The conventional method had difficulties in handling homonyms and multiple location names in a sentence. By using collective intelligence based on the number of co-occurrences among events, person names, location names, and addresses in the Web, our method can effectively overcome these difficulties. Next, we proposed a method for evaluating the impacts of historical entities. We extended the PageRank algorithm to calculate the temporal and spatial impacts of entities. Finally, we showed that our new application "Virtual History Tour" enables users to browse historical events through timeline and map interfaces. Our future work includes incorporating the temporal and spatial impacts of entities into history tours. We also plan to extend our system to other languages besides Japanese to see if it works.

Acknowledgements

This work was supported in part by the following projects and institutions: Grants-in-Aid for Scientific Research (Nos. 18049041, 21700105, and 21700106) from MEXT of Japan, a Kyoto University GCOE Program entitled "Informatics Education and Research for Knowledge-Circulating Society," and the National Institute of Information and Communications Technology, Japan.

References

1. Angel, A., Lontou, C., Pfoser, D.: Qualitative geocoding of persistent web pages. In: Proc. of the 16th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems (GIS 2008), pp. 1–10 (November 2008)
2. Borges, K.A.V., Laender, A.H.F., Medeiros, C.B., Davis Jr., C.A.: Discovering geographic locations in web pages using urban addresses. In: Proc. of the 4th ACM Workshop on Geographical Information Retrieval (GIR 2007), pp. 31–36 (November 2007)
3. Brin, S., Page, L.: The anatomy of a large-scale hypertextual web search engine. In: Proc. of the 7th International Conference on World Wide Web (WWW 1998), pp. 107–117 (April 1998)
4. Garera, N., Yarowsky, D.: Structural, transitive and latent models for biographic fact extraction. In: Proc. of the 12th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2009), pp. 300–308 (March 2009)
5. Gey, F., Shaw, R., Larson, R., Pateman, B.: Biography as events in time and space. In: Proc. of the 16th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems (GIS 2008), pp. 1–2 (November 2008)
6. Haveliwala, T.H.: Topic-sensitive PageRank: A context-sensitive ranking algorithm for web search. IEEE Transactions on Knowledge and Data Engineering (TKDE 2003) 15(4), 784–796 (2003)

7. Jatowt, A., Kanazawa, K., Oyama, S., Tanaka, K.: Supporting analysis of future-related information in news archives and the web. In: Proc. of the 9th ACM/IEEE-CS Joint Conference on Digital Libraries (JCDL 2009), pp. 115–124 (June 2009)
8. Kimura, R., Oyama, S., Toda, H., Tanaka, K.: Creating personal histories from the web using namesake disambiguation and event extraction. In: Baresi, L., Fraternali, P., Houben, G.-J. (eds.) ICWE 2007. LNCS, vol. 4607, pp. 400–414. Springer, Heidelberg (2007)
9. Larson, R.R.: Geographic information retrieval and spatial browsing. In: GIS and Libraries: Patrons, Maps and Spatial Information, pp. 81–124 (April 1996)
10. Larson, R.R.: Geographic ir and visualization in time and space. In: Proc. of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2008), p. 886 (July 2008)
11. Larson, R.R., Frontiera, P.: Geographic information retrieval (gir): Searching where and what. In: Proc. of the the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2004), p. 600 (July 2004)
12. Larson, R.R., Shaw, R.: Mapping life events: Temporal and geographic context for biographical information. In: Proc. of the 9th ACM/IEEE-CS Joint Conference on Digital Libraries (JCDL 2009), pp. 471–472 (June 2009)
13. Liu, X., Nie, Z., Yu, N., Wen, J.R.: Biosnowball: Automated population of wikis. In: Proc. of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD 2010), pp. 969–978 (July 2010)
14. Oyama, S., Shirasuna, K., Tanaka, K.: Identification of time-varying objects on the web. In: Proc. of the 8th ACM/IEEE-CS Joint Conference on Digital Libraries (JCDL 2008), pp. 285–294 (June 2008)
15. Paşca, M.: Organizing and searching the world wide web of facts – step two: Harnessing the wisdom of the crowds. In: Proc. of the 16th International Conference on World Wide Web (WWW 2007), pp. 101–110 (May 2007)
16. Paşca, M., Lin, D., Bigham, J., Lifchits, A., Jain, A.: Organizing and searching the world wide web of facts – step one: the one-million fact extraction challenge. In: Proc. of the 21st National Conference on Artificial Intelligence (AAAI 2006), pp. 1400–1405 (July 2006)
17. Petras, V., Larson, R.R., Buckland, M.: Time period directories: A metadata infrastructure for placing events in temporal and geographic context. In: Proc. of the 6th ACM/IEEE-CS Joint Conference on Digital Libraries (JCDL 2006), pp. 151–160 (June 2006)
18. Stoev, S.L., Feurer, M., Ruckaberle, M.: Exploring the past: a toolset for visualization of historical events in virtual environments. In: Proc. of the the ACM Symposium on Virtual Reality Software and Technology (VRST 2001), pp. 63–70 (November 2001)
19. Strötgen, J., Gertz, M., Popov, P.: Extraction and exploration of spatio-temporal information in documents. In: Proc. of the 6th ACM Workshop on Geographic Information Retrieval (GIR 2010), pp. 1–8 (February 2010)

Investigation of Recommendation Method Considering Region-Restrictedness of Spots

Kenta Oku and Fumio Hattori

College of Information Science and Engineering, Ritsumeikan University,
1-1-1 Nojihigashi, Kusatsu-city, Shiga, Japan
oku@fc.ritsumei.ac.jp, fhattori@is.ritsumei.ac.jp

Abstract. We propose and evaluate a recommendation method that considers region-restrictedness. In our previous work, we defined a spot as an establishment such as a restaurant, amusement facility, or tourist attraction in the real world. A spot with high region-restrictedness indicates that the spot is located in a restricted area but not in a user's home area. We defined the region-restrictedness score to extract region-restricted phrases from text data about spots (such as promotional descriptions about spots). Then, spots including these phrases are recommended to the user. In this paper, we present our proposed method and discuss it on the basis of quantitative and qualitative experimental results.

Keywords: Recommendation, Region-restrictedness, Local search.

1 Introduction

In recent years, many users have begun using local search services to search for spots matching queries in a specific area. Google Maps[1] and Yahoo! Maps[2] are examples of such local search services. When users search for an area using an address or a place name along with keywords such as "bar" or "lunch," close spots matching the keywords are shown on the map.

However, simply providing information about the close spots does not always satisfy the users. For example, suppose Ken, who lives in Osaka, Japan, goes on a trip to Matsusaka, Mie prefecture, Japan. At night, he searches for restaurants close to Matsusaka station and the foods available at these restaurants using a local search service. Although he can obtain information about many restaurants close to the station, some of the restaurants listed are also available in Osaka; for example, Ken can eat a "teriyaki burger" at "McDonald's" or a "hamburg steak" at "Gusto" (a chain of family restaurants in Japan). Because Ken has been eagerly awaiting this trip, he might not feel like going to these restaurants knowing that they are also available in Osaka. A better solution would be to provide Ken with information about restaurants and food available in Matsusaka but not in Osaka (e.g. "Matsusaka beef" (a well-known food available in Matsusaka)).

In case of "Matsusaka beef," a well-known specialty, Ken can retrieve information using keyword queries because he can easily associate "Matsusaka" with

K. Tanaka, P. Fröhlich, and K.-S. Kim (Eds.): W2GIS 2011, LNCS 6574, pp. 36–49, 2011.

"Matsusaka beef." However, when he goes on a trip to an unknown place, he would not know the names of special food. In order to find such foods, it is necessary to automatically extract keywords related to the foods available in a restricted area and show them to users.

In our previous work, we proposed a recommendation method for region-restricted spots [3]. We defined a region-restricted spot as a spot that is located in a restricted destination area but not in a user's home area. Here, the destination area implies an area that the user is visiting when on a trip. The user's home area implies an area where the user lives. In this paper we evaluate our method qualitatively and show the experimental results.

The remainder of this paper is organized as follows. In Section 2, we present related works and services and describe the difference between our proposed method and such existing methods. In Section 3, we explain our recommendation method for region-restricted spots we proposed in the previous work. Section 4 presents basic experimental analyses of the validity of our method when applied to real areas in Japan. Finally, Section 5 presents the conclusions of this paper.

2 Related Work

Tezuka et al. [4] have proposed a method for predicting the regionality of Web pages and objects (e.g., "red leaves" or "noodles"). For example, when a user wants to know about places that are famous for red leaves, he/she can input "red leaves" as a search query, and this will display areas related to red leaves. Therefore, the user can find related areas from the names of objects. On the other hand, in this paper, we aim to extract region-restricted phrases based on the input area.

Tarumi et al. [5] proposed the SpaceTag system to provide information to users in a restricted area. The SpaceTag is an object such as text, an image, a sound, or a program that is accessible in a restricted time and an area. A SpaceTag is made by enterprises, public organizations, and general users. Therefore, their opinions and intentions might be included in the SpaceTag. On the other hand, our proposed method automatically extracts region-restricted phrases from a large amount of spot data associated with real-world locations.

Many location- and time-based information delivery systems for tourism have also been proposed [6][7][8]. However, these systems provide information about only those spots that are close to a user's current location. In other words, they do not aim at region-restrictedness.

Many gourmet Web sites provide information about restaurants that serve local specialties. However, such sites might also include the owner's subjective intentions. Moreover, although sufficient information is available about well-known places, little or no information is available about unknown places. On the other hand, if spot data were to be associated with real-world locations, our proposed method can be applied to any place, including unknown ones. Furthermore, because our method automatically extracts region-restricted phrases, unexpected phrases could also be discovered, such as minor phrases that a site owner might not notice.

3 Recommendation Method Considering Region-Restrictedness

In this section, we explain our proposed recommendation method that considers region-restrictedness.

Our proposed method focuses on the region-restrictedness of spots, and it recommends spots with high region-restrictedness. In this study, we define a spot as an establishment such as a restaurant, amusement facility, or tourist attraction in the real world. A spot with high region-restrictedness indicates a spot that is located in a restricted area but not in a user's home area.

We can acquire spot data such as location and other information from spot information sites such as Gournavi[9]. The proposed method utilizes the following information as spot data:

- Spot name
- Location (latitude/longitude or an address)
- Text data such as promotional descriptions

The proposed method recommends spots based on the following steps:

i Acquire home spots and destination spots.
ii Extract phrases from text data about each destination spot.
iii Calculate region-restrictedness score of each extracted phrase.
iv Recommend spots based on the region-restrictedness score.

In the remainder of this section, we explain each step.

3.1 Acquire Home Spots and Destination Spots

First, we define a home spot and a destination spot (see Figure 1).

We define a home spot h_i as a spot in a user's home area. A home spot set is represented as follows:

$$H = \{h_1, h_2, \ldots, h_n\} \tag{1}$$

Here, n is the total number of home spots. For example, if a user's home area is "Osaka prefecture," the home spot set H includes all spots in "Osaka prefecture" with n being the number of spots.

We define a destination spot l_j as a spot in a user's destination area (e.g., area visited for tourist or business purposes). A destination spot set, which includes spots existing within a radius of r from a base point in the destination area, is represented as follows:

$$L = \{l_1, l_2, \ldots, l_m\} \tag{2}$$

Here, m is the number of spots existing within the range. For example, for the destination "Matsusaka" with the base point as "Matsusaka station," the destination spot set L includes all spots within a radius of r from "Matsusaka station" with m being the number of spots.

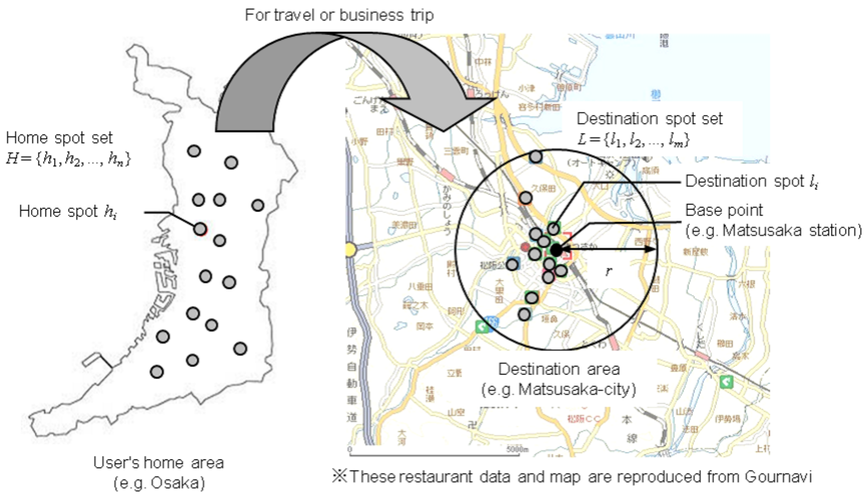


Fig. 1. Home spots and destination spots

Table 1. Extracted patterns of a part of speech

Phrase	Part of speech	# of the parts
Prefix	prefix(+noun)	0 or 1
	Noun(verbal)	0 and over
Base word	Noun(general) or Noun(proper)	1 and over
Suffix	Noun(verbal)	0 and over
	Noun(suffix)	0 or 1

3.2 Extract Phrases from Text Data of Each Destination Spot

Our method extracts phrases included in text data about each destination spot $l_j \in L$. We can use a morphological parser such as ChaSen[10] to extract these phrases. Phrases are extracted by the following steps:

- i Extract words included in text data about each spot l_j using morphological parser.
- ii Let a basic word be a word whose part of speech is "noun (general)" or "noun (proper)" among the extracted words. If "noun (general)" or "noun (proper)" occurs before or after the basic word in a sequence, combine these words into one basic word. This prevents the loss of their features that may occur by splitting them, for example, "Matsusaka beef" into "Matsusaka" and "beef." On the other hand, exclude a basic word consisting of only "noun (proper.place)." Although the method may extract a name of an area such as "Matsusaka" as a region-restricted phrase, in this case, "Matsusaka" is obvious to the user.

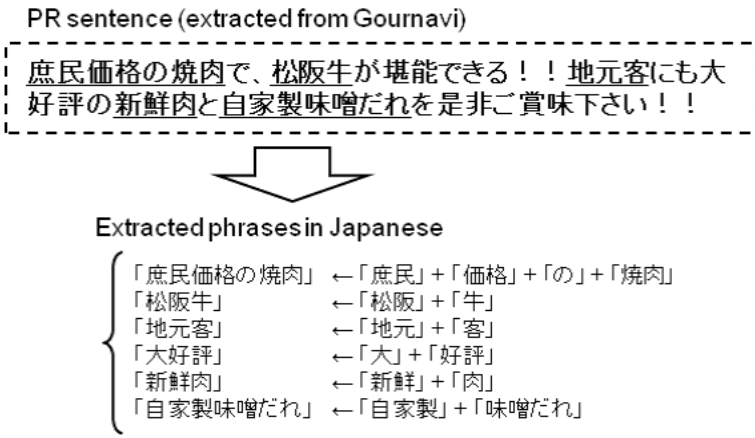


Fig. 2. A Japanese example of the extraction of phrases from text data about a sample spot

iii Combine the basic word and a prefix and suffix dependent on it into one phrase. In other words, extract a noun phrase consisting of patterns of a part of speech shown in Table 1.

iv In the case of a noun phrase that can be combined by "particle (pronominal)," combine these words into one phrase. In this manner, the method can extract more restricted phrase such as "well-established restaurant in Matsusaka-city" or "store with Sukiya architecture in a purely Japanese style."

In Figure 2, we show an example of the extraction of phrases from text about a spot. For example, a phrase "grilled meat with unpretentious price" was extracted by combining "unpretentious," "price," "with ()," and "grilled meat."

We represent an extracted phrase set for a spot l_j as follows:

$$W_j = \{w_{j1}, w_{j2}, \ldots\} \tag{3}$$

3.3 Calculate Region-Restrictedness Score of Extracted Phrase

Our method examines how each extracted phrase $w_{jk} \in W$ is restricted to the destination area. In order to find it, we apply Inverse Document Frequency (IDF), which is widely used in the field of document retrieval. Generally, the IDF is represented as follows:

$$\mathrm{idf} = \log \frac{d}{d_e} \tag{4}$$

Here, d is the total number of documents and d_e, the number of documents including a word w_e. That is, a lower weight is assigned to a general word that commonly occurs in many documents and a higher weight is assigned to a feature word that occurs in specified documents.

Based on this notion, we define the restrictedness ν_{jk}. The restrictedness denotes how each extracted phrase w_{jk} in text about a destination spot is restricted against a user's home spot set. In equation (4), when we regard the text data of spots as documents, the restrictedness is represented as follows:

$$\nu_{jk} = \log \frac{n+1}{n_{jk}+1} \tag{5}$$

Here, n is the number of spots included in a home spot set H (refer to 3.1) and n_{jk}, the number of spots whose text includes a phrase w_{jk} among the H. By using this restrictedness, our method can extract a restricted phrase that occurs only in the destination spots but not in the user's home spots.

However, by considering only the restrictedness, a unique catch copy about a spot, such as "with salad Vikings of coupon" or "vegetables for alcoholic drinks," may also be extracted. Most phrases used for such catch copy are unique. Therefore, the restrictedness of these phrases is higher. Therefore, it is not necessary that some content has regionality for a destination area.

Therefore, we consider the regional weight γ_{jk} in addition to the restrictedness ν_{jk}. The regional weight γ_{jk} indicates how a phrase w_{jk} is related to the destination area. That is, a higher weight is given to a phrase that is strongly related to the destination area, such as "Matsusaka beef" in "Matsusaka," and a lower weight is given to a phase that is weakly related to one such as "with salad Vikings of coupon."

In order to take the regional weight, we use WebPMI[11], which indicates the similarity between words based on their co-occurrence frequencies on the Web. The WebPMI between words p and q is represented as follows:

$$\mathrm{WebPMI}(p,q) = \begin{cases} 0 & \text{if } H(p \cap q) \le c \\ \log \frac{\frac{H(p\cap q)}{N}}{\frac{H(p)}{N}\frac{H(q)}{N}} & \text{otherwise} \end{cases} \tag{6}$$

Here, $H(p), H(q)$, and $H(p \cap q)$ denote the number of Web search results obtained using the queries "p," "q," and "p + q," respectively. N is the number of documents a search engine indexes. c is a threshold for avoiding noise caused by words with low frequency.

By calculating the WebPMI(w_{jk}, local) between a phrase w_{jk} and the name of a destination area local, the method obtains γ_{jk}, which is the regional weight of the phrase w_{jk} for an area. γ_{jk} is defined as follows:

$$\gamma_{jk} = \mathrm{WebPMI}(w_{jk}, \mathrm{local}) \tag{7}$$

Here, we utilize reverse geocoding [1] to detect the name of a destination area. Let local be the name of a city, such as "Matsusaka city," that has been acquired by reverse geocoding from the latitude and longitude of the base point.

Finally, we obtain the region-restrictedness score s_{jk} of a phrase w_{jk} based on the above two measures, namely, the restrictedness ν_{jk} and the regional weight γ_{jk}. The score is calculated as follows:

$$s_{jk} = \nu^{\star}_{jk} \times \gamma^{\star}_{jk} \tag{8}$$

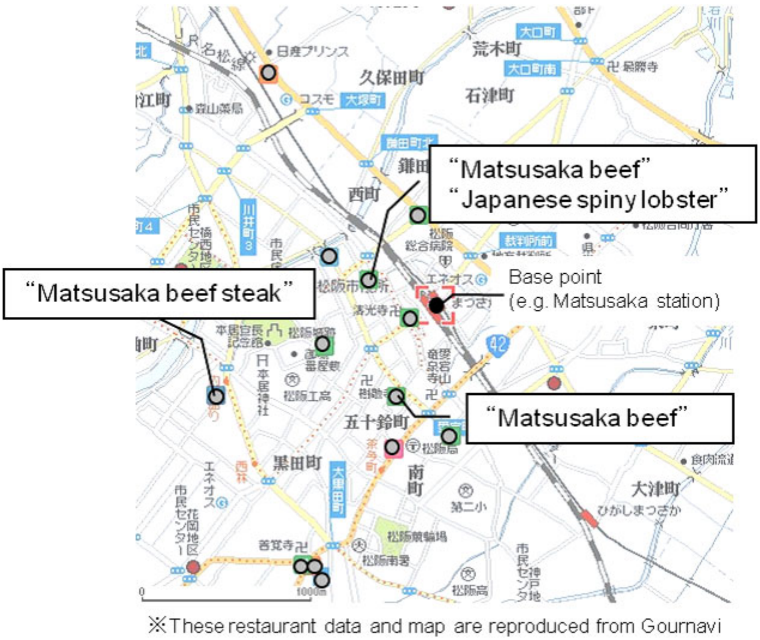


Fig. 3. Example of spot recommendation using region-restrictedness phrases

Here, $\nu^{\star}_{jk}$ and $\gamma^{\star}_{jk}$ are normalized ν_{jk} and γ_{jk} as $[0, 1]$ in spot l_j, and they are respectively calculated as follows.

$$\nu^{\star}_{jk} = \frac{\nu_{jk} - \min\limits_{k}\ \nu_{jk}}{\max\limits_{k}\ \nu_{jk} - \min\limits_{k}\ \nu_{jk}} \qquad \gamma^{\star}_{jk} = \frac{\gamma_{jk} - \min\limits_{k}\ \gamma_{jk}}{\max\limits_{k}\ \gamma_{jk} - \min\limits_{k}\ \gamma_{jk}} \tag{9}$$

3.4 Recommend Spots Based on Region-Restrictedness Score

For the threshold δ, we regard phrases that satisfy $s_{jk} \geq \delta$ as region-restricted phrases. Our recommendation method provides users with spots with these phrases.

Figure 3 shows an example of spot recommendation using region-restricted phrases. For a base point as the user's current or input location, the method shows the surrounding spots with the region-restricted phrases on the map. As shown in Figure 3, our method might extract multiple phrases for one spot or the same phrase for multiple spots. A user can choose spots that he/she wants to use by referring to such phrases.

4 Experiments

We carried out basic experiments in order to analyze the tendency of extraction of phrases based on the region-restricted score described above. First, we explain

Table 2. Example of restaurant data used in this experiment

Data type		Content in Japanese
Restaurant name	-	相生亭
Location	Latitude	34.574997
	Longitude	136.534744
Text data	PR sentence (long)	心あたたまる日本の味覚を『相生亭』にて心ゆくまでお楽しみください。
	PR sentence (short)	【松阪肉会席】極上の松阪肉を、こだわりの素材と相生の味わいでバリエーションも豊かな...

Table 3. Prefectures used as home areas in this experiment

Prefecture	# of restaurant data
Osaka	5,956
Tokyo	5,660
Aichi	3,041

the data sets used in the experiments in Section 4.1. Section 4.2 shows the results of quantitative analysis to evaluate performance of our method. In Section 4.3, we show the results of qualitative analysis to evaluate the relevance of the region-restrictedness score and discuss the results. In Sections 4.4 and 4.5, we discuss the results of extracted phrases depending on the destination spot areas and home areas, respectively.

4.1 Data Set

Although our proposed method can be applied to a restaurant, amusement facility, or tourist attraction, in these experiments, we applied it to restaurants as a genre of spots.

We acquired restaurant data from Gournavi [9], a Japanese restaurant guide. We obtained the following restaurant data using the Gournavi API [9] provided by Gournavi:

– Restaurant name
– Location (as latitude/longitude)
– Text data (short and long versions of promotional descriptions)

Table 2 shows an example of the data.

In these experiments, we set the prefectures listed in Table 3 as home areas. Let a home spot set of each prefecture be all restaurant data acquired by the Gournavi API for each prefecture.

We also set the base points listed in Table 4 as destination areas. Let a destination spot set of each base point be all restaurant data existing within a radius of $3,000$ m from each base point. However, in cases where the number of restaurants exceeded 30, the closest 30 restaurants from the base point were chosen.

4.2 Quantitative Analysis of Region-Restrictedness Score

We carried out a quantitative analysis of the tendency of phrase extraction based on the region-restrictedness score.

Table 4. Base points used as destination areas in this experiment

Base point	Latitude	Longitude	City name	# of spots
Nara Park	34.685454	135.843411	Nara-city	30
Matsusaka station	34.576917	136.535790	Matsusaka-city	10
Ohmihachiman station	35.122875	136.102753	Ohmihachiman-city	30

Table 5. Example of phrase tagging

Phrase	Useful	Found in home area	Regional to dest. area
Matsusaka-city	0	-	-
Matsusaka beef	1	0	1
Steak	1	1	0

In this analysis, let the home area be "Osaka" and the base points in the destination area be "Nara Park," "Matsusaka station" and "Ohmihachiman station." Phrases were extracted from text data about each restaurant data in the destination spot set according to the steps described in Section 3.2. The numbers of extracted phrases were 483, 165 and 265 respectively (except for repeated phrases).

We ranked the extracted phrases in order of the region-restrictedness score. Then participants in this experiment tagged each phrase with whether the phrase was "useful" to show them. For all phrases tagged with "useful," they tagged each phrase with whether the phrase can be "found in the home areas" and the phrase has "regionality to the destination areas". Table 5 shows examples of the phrase tagging in case of "Osaka" as the home area and "Matsusaka station" as the destination area. The participants thought that "Matsusaka-city" was not "useful," while "Matsusaka beef" and "Steak" were "useful." "Steak" can be "found in also Osaka." On the other hand, "Matsusaka beef" has "regionality to Matsusaka." Thus, our method aims to extract phrases tagged with "useful" and "regional to the destination area." Therefore, we regard these phrases as compatible data while the other phrases as incompatible data. Based on these compatible and incompatible data, we obtained precision of our method.

Figure 4 indicates precision every ten ranks. The precision denotes the average of those three cases. We can see from this Figure, the higher ranked phrases have higher precision, while the lower ranked phrases have low precision. Although, this Figure shows precision between 10 and 150 ranks, the lower ranked phrases alsow showed precision of 0. In next section, we show these results in qualitatively.

4.3 Qualitative Analysis of Region-Restrictedness Score

We carried out a qualitative analysis of the tendency of phrase extraction based on the region-restrictedness score in order to evaluate the relevance of the score.

In this analysis, let the home area be "Osaka" and the base point in the destination area be "Nara Park." Phrases were extracted from text data about

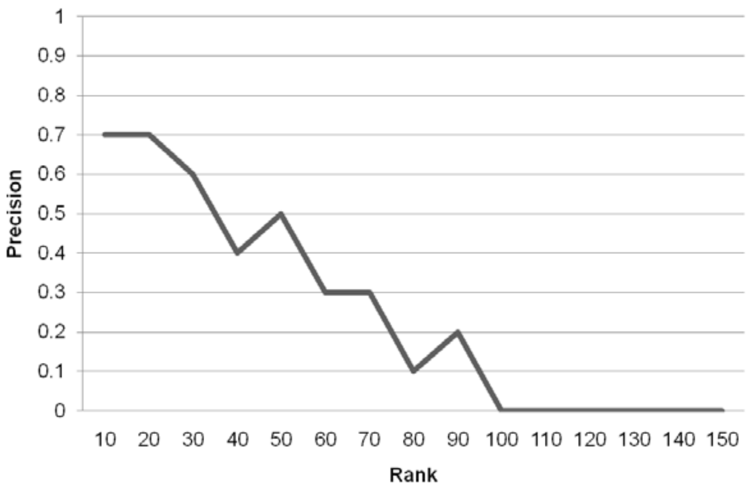


Fig. 4. Average precision every ten ranks

each restaurant data in the destination spot set according to the steps described in Section 3.2. The number of extracted phrases was 483 (except for repeated phrases).

The region-restrictedness score was calculated for each extracted phrase. Then, we constructed a phrase ranking based on the score. Here, let $c = 5$ in Equation (6). The number of phrases that satisfies $H(p \cap q) \leq c$ was 47. We then ranked 436 phrases and excluded these 47 phrases. Figure 5 shows a part of the ranking results. In order to evaluate the relevance of the ranking, we showed the following:

(a) Top 30: 1st to 30th
(b) Worst 30: 407th to 436th (the lowest)

In the next section, we discuss each result.

Discussion of Top 30 Phrases. As shown in Figure 5 (a), we found that many phrases that were strongly related to Nara but not to Osaka, such as "Yamato chicken" (9th), "Yoshino Kudzu" (13th), "right in the middle of Nature in Mt. Kasuga" (1st), and "Todaiji" (14th) were extracted. We focused on the contents of extracted phrases, and found that the phrases could be broadly categorized as follows.

A. Phrases related to foods or ingredients
B. Phrases characterizing a restaurant in terms of its ambience or facilities
C. Phrases including a name of a landmark around a restaurant
D. Phrases related to a restaurant name

Each alphabet in Figure 5 (a) indicates one of the above categories. We now discuss each category.

(a) Top 30

Rank	Phrase in English	Score	Category
1	Right in the middle of Nature in Mt. Kasuga	1.0000	C
2	The middle of famous temples	0.9800	C
3	Hirokuni Akiyoshi (a chef's name at a hotel in Nara)	0.9756	B
4	Inn of ancient city	0.9687	B
5	Garden bonito	0.9655	A
6	Kasuga Grand Shrine (a Shinto shrine in Nara Park)	0.9158	C
7	Nara Park (a public park located in the city of Nara)	0.9151	C
8	Specially made bonito	0.9094	A
9	Yamato chicken (a traditional chicken in Nara)	0.9032	A
10	Dining bar open	0.8980	B
11	French terrace	0.8970	B
12	Woody	0.8910	B
13	Yoshino Kudzu (a traditional specialty in Nara)	0.8852	A
14	Todai-ji (a Buddhist temple complex located in Nara Park)	0.8781	C
15	Town of Nara	0.8723	B
16	In Higashimuki shopping street (a shopping street in Nara)	0.8592	C
17	Kintetsu Nara station	0.8468	C
18	Fastidious bonito	0.8440	A
19	Kudara (a restaurant name)	0.8196	D
20	Primeval forest in Kasuga Okuyama (one of the world heritages in Nara)	0.8118	C
21	Front of Kintetsu Nara station	0.8100	C
22	Vicinity of Sarusawa pond (a pond in Nara Park)	0.8046	C
23	Terrace style	0.7843	B
24	Yamato vegetables (traditional vegetables in Nara)	0.7798	A
25	Yamato beef (a traditional beef in Nara)	0.7738	A
26	Sister restaurant to beer hall Fujin (a restaurant name)	0.7735	D
27	Japanese-style restaurant inn	0.7626	D
28	Banzai (a restaurant name)	0.7599	D
29	Kimono fabric	0.7560	B
30	Japanese-style restaurant Miyama (a restaurant name)	0.7266	D

(b) Worst 30

Rank	Phrase in English	Score
407	Reserved	0.1021
408	Mind	0.1008
409	Popular	0.0991
410	Drink	0.0978
411	Woman	0.0969
412	Lunch at restaurant	0.0928
413	This restaurant	0.0920
414	Taste	0.0876
415	Summer	0.0870
416	Wine	0.0856
417	Ingredient	0.0770
418	Alcoholic drink	0.0767
419	Maximum	0.0766
420	Banquet	0.0751
421	Other wine	0.0746
422	Atmosphere	0.0744
423	Farewell and welcome party	0.0727
424	Walking	0.0681
425	Pace of a month	0.0615
426	Customer	0.0589
427	Space	0.0581
428	Menu	0.0575
429	Location by walking	0.0526
430	Dinner party	0.0475
431	Economical	0.0400
432	Private separate room	0.0372
433	Inside of this restaurant	0.0316
434	Restaurant	0.0227
435	As much as one likes	0.0178
436	Course	0.0000

Fig. 5. Ranking results of phrases based on region-restrictedness scores (home: Osaka, destination: Nara Park)

A. Phrases related to foods or ingredients

Phrases related to foods or ingredients, such as "Yamato chicken" (9th), "Yoshino Kudzu" (13th), and "Yamato beef" (25th), that were peculiar to Nara were assigned higher ranks.

However, although "tea gruel" and "kakinoha sushi" are also specialties peculiar to Nara, these phrases could not be extracted in this experiment because they were not included in the promotional sentences used as text data about restaurants. On the other hand, some restaurant data include these phrases in their details pages or menu pages. Therefore, we would like to consider these pages as text data about spot data in future work.

B. Phrases characterizing a restaurant in terms of its ambience or facilities

"Inn of ancient city" (4th) and "woody" (12th) were phrases that characterized the restaurant's ambience, and they were assigned higher ranks. "French terrace" (11th) and "terrace style" (23th) were phrases that characterized the restaurant's facilities and "Hirokuni Akiyoshi" (3rd) was the name of a chef; these phrases were also assigned higher ranks.

Using such phrases, a user can choose restaurants by also referring to the ambience or facilities peculiar to the destination area.

However, "French terrace" and "terrace style" are not region-restricted phrases for Nara alone. These phrases were extracted because there were a few samples in Osaka that included these phrases in the promotional sentences. In the future, we intend to examine whether it is better to consider spots available not in a home area or only in a destination area.

(a) Destination: Matsusaka station

Rank	Phrase in English	Score
1	Matsusaka merchant (a traditional merchant in Matsusaka)	1.0000
2	Gojoban Yashiki (a historic residence in the city of Matsusaka)	0.9828
3	Matsusaka beef (a specialty in Matsusaka) etc.	0.9547
4	Matsusaka meat (a specialty in Matsusaka)	0.9298
5	Old - established restaurant in Matsusaka-city	0.8169
6	Old - established restaurant times	0.8039
7	Beef Kaiseki (a traditional multi-course Japanese dinner)	0.7866
8	Look of a store with Sukiya structure in a purely Japanese style	0.7812
9	Yachiyo (a restaurant name)	0.7802
10	Matsusaka beef steak (a specialty in Matsusaka)	0.7348

(b) Destination: Ohmihachiman

Rank	Phrase in English	Score
1	Hachiman Konnyaku (a specialty in Ohmihachiman)	0.9226
2	Natural bittern (a bittern from Ohmihachiman)	0.8948
3	Kansai sushi	0.8690
4	Chojifu (a specialty in Ohmihachiman)	0.8393
5	JR Ohmihachiman station	0.8126
6	Dashimaki with Akaji Tamago (Japanese style omelet with eel)	0.8080
7	Prelude to autumn	0.7869
8	Funazushi (a sushi with a crucian carp)	0.7659
9	Dengoroh (a restaurant name)	0.7604
10	Front of Ohmihachiman station	0.7185

Fig. 6. Ranking results of phrases depending on destination areas (home: Osaka)

C. Phrases including a name of a landmark around a restaurant
"Right in the middle of Nature in Mt. Kasuga" (1st), "the middle of famous temples" (2nd), "Todaiji" (14th), and "primeval forest in Kasuga Okuyama" (20th) were assigned higher ranks as phrases including a name of a landmark around a restaurant. Because most landmarks indicate the regionality, it is also effective to provide such landmarks in addition to the phrases characterizing a restaurant.

D. Phrases related to a restaurant name
"Kudara" (19th) and "Banzai" are the names of restaurants. First, because most restaurant names are peculiar, these phrases can be extracted easily. However, the names do not always characterize their contents. Therefore, it is not effective to show these phrases to users. Therefore, it is necessary to exclude these phrases related to a restaurant name from the ranking.

Discussion of Worst 30 Phrases. Figure 5 (b) shows the worst 30 phrases.

We found that general phrases such as "course" and "restaurant" were assigned lower ranks. As a result, we confirmed that the general phrases could be assigned lower ranks using the region-restrictedness score.

4.4 Discussion of Phrases Depending on Destination Areas

We compared the ranking results of extracted phrases for the home area "Osaka" and destination areas as the base points listed in Table 4. Figure 6 shows the top 10 phrases when the base points were "Matsusaka station" and "Ohmihachiman station." 165 and 265 phrases were extracted for these respective base points.

(a) Home: Tokyo

Rank	Phrase in English	Score
1	Right in the middle of Nature in Mt. Kasuga	1.0000
2	The middle of famous temples	0.9807
3	Hirokuni Akiyoshi (a chef's name at a hotel in Nara)	0.9756
4	Inn of ancient city	0.9687
5	Garden bonito	0.9644
6	Kasuga Grand Shrine (a Shinto shrine in Nara Park)	0.9157
7	Nara Park (a public park located in the city of Nara)	0.9149
8	Specially made bonito	0.9079
9	Yamato chicken (a traditional chicken in Nara)	0.9032
10	Dining bar open	0.8980
11	French terrace	0.8970
12	Woody	0.8920
13	Yoshino Kudzu (a traditional specialty in Nara)	0.8850
14	Todai-ji (a Buddhist temple complex located in Nara Park)	0.8787
15	Town of Nara	0.8725
16	Yamato vegetables (traditional vegetables in Nara)	0.8663
17	In Higashimuki shopping street (a shopping street in Nara)	0.8592
18	Kintetsu Nara station	0.8467
19	Fastidious bonito	0.8370
20	Kudara (a restaurant name)	0.8194

(b) Home: Aichi

Rank	Phrase in English	Score
1	Right in the middle of Nature in Mt. Kasuga	1.0000
2	The middle of famous temples	0.9804
3	Hirokuni Akiyoshi (a chef's name at a hotel in Nara)	0.9767
4	Inn of ancient city	0.9690
5	Garden bonito	0.9626
6	Kasuga Grand Shrine (a Shinto shrine in Nara Park)	0.9167
7	Nara Park (a public park located in the city of Nara)	0.9159
8	Specially made bonito	0.9094
9	Yamato chicken (a traditional chicken in Nara)	0.9024
10	Dining bar open	0.8996
11	French terrace	0.8963
12	Woody	0.8938
13	Yoshino Kudzu (a traditional specialty in Nara)	0.8834
14	Todai-ji (a Buddhist temple complex located in Nara Park)	0.8799
15	Town of Nara	0.8694
16	Yamato vegetables (traditional vegetables in Nara)	0.8675
17	In Higashimuki shopping street (a shopping street in Nara)	0.8610
18	Kintetsu Nara station	0.8472
19	Fastidious bonito	0.8440
20	Kudara (a restaurant name)	0.8192

Fig. 7. Ranking results of phrases depending on home areas (destination: Nara Park)

As shown in Figure 6, phrases restricted to each area could be extracted, such as "Matsusaka beef" at "Matsusaka station" and "Hachiman Konnyaku" at "Ohmihachiman station." In particular, foods based on "Matsusaka beef" such as "Matsusaka beef steak" (10th) were extracted in various phrases at "Matsusaka station" and various foods such as "Hachiman Konnyaku" (1st), "natural bittern" (2nd), and "Chojifu" (4th) were extracted at "Hachiman station"; in addition, various landmarks such as "Kasuga Taisya" and "Todaiji" were extracted at "Nara Park." Therefore, the characteristics tend to depend on the destination areas. In the future, we intend to carry out further detailed analyses using various areas.

4.5 Discussion of Phrases Depending on Home Areas

We compared the ranking results of extracted phrases for the destination area "Nara Park" and home areas as the prefectures listed in Table 3. Figure 7 shows the top 20 phrases when the home areas were "Tokyo" and "Aichi."

As shown in Figure 7, there was not much difference among the three prefectures in this experiment. In only one case, we found that "Yamato vegetable" was ranked 16th in "Tokyo" and "Aichi," whereas it was ranked 24th in "Osaka." Actually, according to data obtained from Gournavi, there were three restaurants serving this food in Osaka but none in Tokyo and Aichi. This implies that users have some opportunities to eat this food in Osaka but none in Tokyo and Aichi. This is why its rank was lower in the case of Osaka than in Tokyo and Aichi. This implies that the ranking results tend to change depending on the home areas.

Although there was not much difference in this experiment, we intend to examine the effect of the home area in greater detail in future work.

5 Conclusion

We proposed a recommendation method that considers region-restrictedness. In this study, we define a spot as an establishment such as a restaurant, amusement facility, or tourist attraction in the real world. A spot with high region-restrictedness indicates a spot that is located in a restricted area but not in a user's home area. Then, our proposed method recommends spots with high region-restrictedness. We define the region-restrictedness score to extract region-restricted phrases for the destination area from text data about spots.

We carried out quantitative and qualitative analyses of the region-restrictedness score. From the ranking results based on the score, we found that many phrases related to destination areas could be extracted with higher ranks. Moreover, we found that the extracted phrases could be broadly categorized into four categories, and we discussed each category separately.

Although we have presented a basic analysis of the recommendation method, we intend to examine this method further with the aim of developing a practical recommendation system that considers region-restrictedness.

References

1. Google maps, `http://maps.google.com/`
2. Yahoo! maps, `http://maps.yahoo.com/`
3. Oku, K., Hattori, F.: Basic study on a recommendation method considering region-restrictedness of spots. In: Yoshikawa, M., Meng, X., Yumoto, T., Ma, Q., Sun, L., Watanabe, C. (eds.) DASFAA 2010. LNCS, vol. 6193, pp. 353–364. Springer, Heidelberg (2010)
4. Tezuka, T., Kondo, H., Tanaka, K.: Estimation of relevant regions for web content by gaussian mixture models for object level local search. Information Processing Society of Japan: Database 1(1), 13–25 (2008) (in Japanese)
5. Tarumi, H., Morishita, K., Kambayashi, Y.: Public applications of spacetag and their impacts, digital cities: Technologies, experiences and future perspectives. In: Ishida, T., Isbister, K. (eds.) Digital Cities 1999. LNCS, vol. 1765, pp. 350–363. Springer, Heidelberg (2000)
6. Abowd, G.D., Atkeson, C.G., Hong, J., Long, S., Kooper, R., Pinkerton, M.: Cyberguide: A mobile context-aware tour guide. Wireless Networks 3(5), 421–433 (1997)
7. Sumi, Y., Etani, T., Fels, S., Simonet, N., Kobayashi, K., Mase, K.: C-map: Building a context-aware mobile assistant for exhibition tours. In: Community Computing and Support Systems, Social Interaction in Networked Communities, pp. 137–154. Springer, London (1998)
8. Cheverst, K., Davies, N., Mitchell, K., Friday, A.: Experiences of developing and deploying a context-aware tourist guide: the guide project. In: Proceedings of the 6th Annual international Conference on Mobile Computing and Networking (MobiCom 2000), pp. 20–31. ACM, New York (2000)
9. Gournavi, `http://www.gnavi.co.jp/en/`
10. Chasen, `http://chasen.naist.jp/hiki/ChaSen/` (in Japanese)
11. Bollegala, D., Matsuo, Y., Ishizuka, M.: Measuring semantic similarity between words using web search engines. In: WWW 2007 (2007)

A Case Study of Participatory Data Transfer for Urban Temperature Monitoring

Houtan Shirani-Mehr[1], Farnoush Banaei-Kashani[1], Cyrus Shahabi[1], and Lin Zhang[2]

[1] University of Southern California, Los Angeles, CA
{hshirani,banaeika,shahabi}@usc.edu
[2] Tsinghua University, Beijing, China
linzhang@tsinghua.edu.cn

Abstract. The sensing systems that monitor physical environments rely on communication infrastructures (wired or wireless) to collect data from the sensors embedded in the environment. However, in many urban environments pre-existing communication infrastructures are not available, and installing new infrastructures is unjustifiably expensive and/or technically infeasible. For such environments, we envision *Participatory Data Transfer* (*PDT*) as an alternative communication medium that leverages users participation for data transfer. With PDT, users use mobile devices to receive data from sensors, and forward the sensed data through the ad hoc network of the mobile devices until the data is received by the data aggregators (i.e., data sinks). Sensor deployment and ad hoc routing/networking are two related problems that are both extensively studied in the literature. However, to enable efficient deployment of PDT for sensing applications one needs to consider the requirements of the two aforementioned problems in conjunction. In this paper, we present a case study of PDT with which we explore the performance of PDT-based data transfer with a sample urban sensing application, namely, an urban temperature monitoring application. Our experimental case study is by simulation based on real datasets including GPS track data for more than 2000 vehicles in the city of Beijing. We discuss our observations based on this case study which can serve as directions to design application-specific optimal PDT mechanisms.

Keywords: Participatory Data Transfer, Sensor Placement, Aggregator Placement, Urban Temperature Monitoring.

1 Introduction

With the recent technological advancements in developing low-power and inexpensive sensing devices, sensing systems that allow for real-time and accurate monitoring of physical environments are becoming prevalent [3,7]. In particular, there are numerous sensing applications for *urban* environment monitoring, such as exposure analysis (pollution, noise, etc.), hazard detection (e.g., chemical contamination, fire, flood), and urban traffic analysis (for vehicles and people). Such

K. Tanaka, P. Fröhlich, and K.-S. Kim (Eds.): W2GIS 2011, LNCS 6574, pp. 50–63, 2011.

sensing systems consist of a set of sensors that sense the urban environment, a set of aggregators (or sinks) where the sensed data is collected for further processing, and a communication infrastructure that enables data transfer between sensors and aggregators. For example, with an urban temperature monitoring application a set of temperature sensors are placed in an environment. The measurements from the sensors are transferred through a wireless network to a set of processing nodes (i.e., aggregators). At the aggregators, the temperature for the entire environment is estimated based on the measurements obtained from the sensors.

While in some urban environments sensing systems can use the existing wired or wireless (e.g., Wi-Fi) infrastructures for communication, in many other environments pre-existing communication infrastructures are not available and installing new infrastructures is unjustifiably expensive or infeasible. For such environments, we envision *participatory data transfer* (*PDT*) [15] as an alternative communication medium with which data is transferred by objects (individuals, vehicles, etc.) that are moving through the environment and are equipped with mobile devices such as cell phones, laptops and PDAs. With PDT, participating objects use their mobile devices to receive data from sensors. Thereafter, the data is forwarded through the ad hoc network of objects' mobile devices until the data is received by the aggregators of the sensing system.

The design objectives for *Quality-aware* PDT (Q-PDT) are to maximize the coverage of the area monitored by the sensing application while minimizing the delay in reporting the sensed phenomenon over time. Therefore, one needs to consider solving two optimization problems conjointly to develop an efficient Q-PDT solution: 1) placement/deployment of the sensors and aggregators given the distribution and movement patterns of the urban population, and 2) routing of the data through the network of the users participating in PDT. Although sensor deployment and ad hoc routing/networking (or opportunistic networking) are both extensively studied in the literature, a Q-PDT solution based on isolated solutions for each of these problems may be poor and ineffective.

In this paper, we present a case study of Q-PDT with which we develop and try various solutions for the two problems of placement and routing in combination to make observations on their joint performance for future design. The experimental case study is by simulation based on real datasets including the GPS tracks of more than 2000 vehicles in the city of Beijing. The vehicles' GPS tracks cover an area of approximately $600km^2$.

The organization of the paper is as follows. In Section 2 we present a model for PDT based sensing applications. Section 3 formally defines the Q-PDT problem. We study the computational complexity of the problem in Section 4. In Section 5, we provide our case study. Section 6 reviews the related work and finally we conclude the paper and discuss our future directions in Section 7.

2 System Model

A PDT based sensing system comprises of a set of sensors and aggregators, and leverages PDT to transfer data from sensors to aggregators to be processed or further forwarded to a central processing unit via some existing communication infrastructure (wired or wireless). With such a system, participating objects collectively serve as a data transfer medium to transfer data from sensors to aggregators. In this section, we present our models for the sensing application as well as the PDT process in a PDT-based sensing system.

2.1 Application Model

The sensing application tracks a phenomenon such as temperature in an environment E based on the measurements collected from the sensors placed in E. Assume the set of points in E is denoted by V. We assume the application monitors the phenomenon in E during the time interval T. T is discretized into equal length intervals, i.e., T=$(I_1, I_2, \ldots, I_n)$, and the application reports the phenomenon variations for every point in V at the end of each I_i. For example, with $|I_i| = 1$ hour for a temperature monitoring application, the application reports hourly the temperature at each point in V. The phenomenon variations during each interval I_i is assumed to be negligible and therefore any reading for a point q during the interval I_i, is a sufficiently accurate representation for the phenomenon during the entire I_i.

We presume the application can place at most N_S sensors and N_A aggregators in E. The set of sensors are denoted by S=$\{s_1, s_2, \ldots, s_{N_S}\}$, and the set of aggregators by A=$\{a_1, a_2, \ldots, a_{N_A}\}$. Accordingly, each sensor $s_i \in S$ generates a series of packets, $P_{s_i}(t)$, where each packet contains a sensor measurement at t. A packet p is considered received if at least one aggregator receives it. We denote the earliest time at which p reaches an aggregator by t_{r_p}. The packet also has a lifetime of T_p which means that the sensors readings carried by packets become stale for real-time monitoring purposes if received after T_p. If p is generated at time $t_p \in I_i$, T_p=$|t_{e_{I_i}} - t_p|$ where $I_i = [t_{b_{I_i}} \;\; t_{e_{I_i}}]$. We assume a sensor s_i generates a packet p whenever an object o can communicate with s_i (while respecting the communication constraints such as distance with s_i). However, if o carries a non-stale packet already generated at s_i, no new packet will be generated to avoid unnecessary data transfer and reduce the communication overhead. Imagine an urban temperature monitoring as the running example. Figure 1 shows three temperature sensors s_1, s_2 and s_3 and two aggregators a_1 and a_2 deployed by this application that are placed in the environment E. If this application reports the hourly temperature of the urban environment (starting at 12:00am), the packets generated at 12:30pm and 13pm have the life times of 30 minutes and one hour, respectively.

2.2 PDT Model

PDT is the process in which a set of objects $O = \{o_1, o_2 \ldots, o_{N_O}\}$ participate in data transfer. The objects may be constrained in their movement in the environment. For example, in Figure 1 four vehicles o_1, o_2, o_3 and o_4 are participating in PDT. The vehicles are moving on a road network R where the road network segments are highlighted in the figure. The vehicles use their embedded mobile devices to exchange packets among each other (For example, o_2 and o_3 exchange

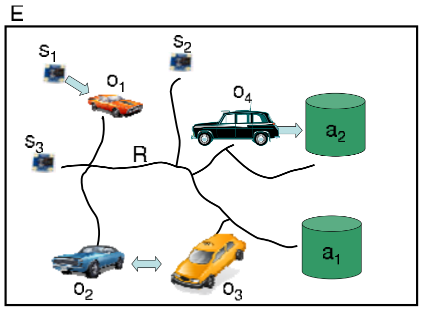


Fig. 1. Sensors, aggregators and objects contribute to PDT

packets collected on their way in Figure 1. Later, each of these vehicles may meet an aggregator and deliver the exchanged packets) and exchange packets with sensors (e.g., o_1 and s_1 in Figure 1) and aggregators (e.g., o_4 and a_2 in Figure 1). A data packet can be transmitted from a device to another one if the devices are *reachable* from each other. Two devices are reachable if they are able to communicate data given the existing communication media. For example, if no wireless or wired connection exists, two cell phones are reachable when they are close enough to exchange data via Bluetooth or infrared. Finally, we assume all communicating devices which are used to exchange data are programmed to participate in PDT automatically; hence, participation does not require active intervention of the users.

3 Problem Definition

In this section, we define the problem of Quality-Aware PDT (Q-PDT). The objective with the Q-PDT is to consider movement patterns of the participating objects and accordingly place a set of sensors and aggregators in an environment such that we can maximize the quality of the packets received by the aggregators during the time interval T. Next, we first formally define the quality of a set of transferred packets, and afterward formalize the Q-PDT problem.

Data Quality: The application reports the phenomenon in the environment E at the end of each $I_i \in T = (I_1, I_2, \dots, I_n)$. Therefore, we define the quality $Q(P)$ of transferred (and collected) packets P over the entire T as the summation of the quality of packets collected during each $I_i \in T$ (we exclude the packets from P which are not received by aggregators during their life-time). In other words,

$$Q(P) = \sum_{i=1}^{n} Q_i(P_i),$$

where P_i is the set of packets received during I_i and $Q_i(P_i)$ represents the quality of packets in P_i. We define $Q_i(P_i)$ as the amount of uncertainty reduction in predicting the phenomenon at unsensed locations (locations where no sensor is placed) during I_i, after receiving packets in P_i. We assume that the sensors report the phenomenon at the sensed locations (locations where sensors are placed) accurately and with no uncertainty. We use Entropy function [12] to measure the uncertainly in predicting the phenomenon at the environment points. Consider a random vector with values in $\mathbb{R}^m$ and probability density function $f(x)$. The entropy of X is defined by

$$H(X) = -\int_{\mathbb{R}^m} \log f(x) dx$$

Assume the set of environment points is represented by V and the set of points where sensors in S_i are located is denoted by V_i. Therefore, to calculate $Q_i(P_i)$ we only need to measure the uncertainty in predicting the phenomenon at unsensed locations. Accordingly, we define

$$Q_i(P_i) = H(V - V_{S_i}) - H(V - V_{S_i} | V_{S_i}), \tag{1}$$

where $H(V - V_{S_i})$ represents the entropy in predicting the phenomenon at points in $V - V_{S_i}$ before placing the sensors. Correspondingly, $H(V - V_{S_i} | V_{S_i})$ is entropy at the same locations after placing sensors at points in V_{S_i}.

However, we need to consider a model for the sensed phenomenon in order to calculate the entropy of predicted phenomenon. A conventional approach for modeling continuous physical phenomenon such as temperature is to assume that sensor readings have (multivariate) Gaussian joint distribution [9,4]. In particular for the sensors in S_i, the set of $n = |S_i|$ corresponding random variables X have Gaussian joint distribution if

$$P(X = x) = \frac{1}{(2\pi)^{n/2}|\Sigma|} e^{-\frac{1}{2}(x-\mu)^T \Sigma^{-1}(x-\mu)},$$

where μ, Σ and $|\Sigma|$ are the mean vector, covariance matrix and determinant of the covariance matrix, respectively. If we index each random variable $X_i \in X$ by i, we end up with a set of indices $\mathcal{V}$. Any subset of random variables with index set $\mathcal{A} \in \mathcal{V}$ has a Gaussian joint distribution as well. In the temperature monitoring application, the goal is to monitor temperature for every point in V and not just points in V_{S_i}. Fortunately, *Gaussian Process* (GP), which is a generalization of multivariate Gaussian, considers an infinite number of random variables and hence can be leveraged to model temperature for all the points in V. A GP is associated with a mean function $\mathcal{M}(.,.)$ and a symmetric covariance function $\mathcal{K}(.,.)$. For each random variable with index $u \in \mathcal{V}$, its mean μ_u is given by $\mathcal{M}(u)$. Accordingly, for each pair of indices $u, v \in \mathcal{V}$, their covariance σ_{uv} is $\mathcal{K}(u, v)$. With GP, given a set of sensor readings $x_\mathcal{A}$ corresponding to set $\mathcal{A} \in \mathcal{V}$, we can predict temperature at any point $y \in \mathcal{V}$ conditioned on $x_\mathcal{A}$, i.e., $P(X_y | x_\mathcal{A})$. The distribution of X_y is Gaussian whose conditional variance $\sigma^2_{y|\mathcal{A}}$ is

$$\sigma^2_{y|\mathcal{A}} = \mathcal{K}(y, y) - \Sigma_{y\mathcal{A}} \Sigma^{-1}_{\mathcal{A}\mathcal{A}} \Sigma_{\mathcal{A}y},$$

where $\Sigma_{y\mathcal{A}}$ is a covariance vector with one entry for each $u \in \mathcal{A}$ with value $\mathcal{K}(y, u)$. The entropy of a Gaussian random variable Z conditioned on a set of variables X is a function of its variance,

$$H(Z|X) = \frac{1}{2}\log(2\pi e\sigma^2_{Z|X}).$$

Consequently, this enables us to calculate $Q_i(P_i)$ based on Equation (1).

Definition 1 (Problem of Quality-Aware PDT (Q-PDT)). *We define the problem of Q-PDT as follow: Given a set of sensors S and a set of aggregators A, place all the sensors and aggregators in an environment E, such that $Q(P)$ is maximized.*

4 Complexity

In this section, we prove that Q-PDT is an NP-hard problem by reduction from the *optimal coverage* problem [8] (which is also an NP-hard problem). The optimal coverage problem can be formalized as follows.

Definition 2 (Optimal Coverage). *Assume an environment E is represented by a discrete set of points V. Given k sensors to monitor a phenomenon in E, optimal coverage places sensors in a set of points $F \subseteq V, |F| = k$, such that the prediction accuracy of the phenomenon throughout E is optimized:*

$$\arg\max \ H(V - F) - H(V - F|F) = F, \tag{2}$$

where the set F is a set of points which maximally reduces the entropy over the rest of the space $V - F$.

The following theorem proves that the Q-PDT problem is NP-hard.

Theorem 1. *Q-PDT is an NP-hard problem.*

Proof. We prove the theorem by providing a polynomial time reduction from the optimal coverage problem. Towards that end, we prove that given an instance of optimal coverage, denoted by I_C, there exists an instance of Q-PDT, denoted by I_Q, such that the solution to I_Q can be converted to the solution of I_C in polynomial time. Assume a given instance I_C whose goal is to place k sensors to monitor a phenomenon during the time interval T'. We propose the following mapping from I_C to I_Q to reduce I_C to I_Q. The environments for both instances are the same, i.e., the set of environment points in I_C are mapped to environment points in I_Q. Furthermore, for both instances we monitor the phenomenon within the same time interval, i.e., T' is mapped to T. We assume there are k pairs of sensors and aggregators in I_Q where the ith pair consists of a sensor s_i and an aggregator a_i such that a_i is placed in the environment with a negligible distance of $\epsilon \approx 0$ with s_i. Due to the small distance between sensors and aggregators in each pair, there is no need for objects to transfer packets from sensors to aggregators. Given this mapping, it is easy to observe that if the answer to I_Q is the set of environment points F, F is also the answer to I_C.

We conclude from Theorem 1 that optimal solution for the Q-PDT problem is rendered unscalable as the spatial extent of the environment grows large. Hence, heuristic solutions should be developed to solve Q-PDT for large environments.

5 Case Study

In this section, we focus on a special use case of Q-PDT in which PDT is deployed in a temperature monitoring application for an urban environment. With this case study, participating objects in PDT are a set of moving vehicles which are constrained to move on a road network R embedded in the environment E. Below, we first explain our experimental setup for this case study and thereafter provide our results and observations.

5.1 Experimental Setup

In this section, we first describe our PDT specifications and our assumed model for distribution of temperature in the environment. Thereafter, as Q-PDT is an NP-hard problem, we propose various heuristics for sensors and aggregators placement. Finally, we describe data routing techniques we used for our case study.

5.1.1 PDT Specifications

We conducted our experiments using real data capturing the movements of vehicles in the city of Beijing. This dataset covers the GPS tracks of more than 2000 distinct vehicles collected during a day. The vehicles' GPS tracks cover an area of approximately $600km^2$. This area comprises the environment E. The vehicles locations are recorded every minute (a total of more than $1,400,000$ records). We assume the vehicles use Dedicated Short-Range Communications (DSCR) channels to communicate with each other, as well as with the sensors and aggregators. The DSCR [18] standard is developed to support low-latency wireless data communications between vehicles and from vehicles to roadside units. The effective range for communication over DSRC is up to 1km. Therefore, two vehicles (or a vehicle and an aggregator/sensor) are reachable if they are at most 1km apart.

5.1.2 Temperature Modeling

Since we did not have access to the temperature data for the city of Beijing (the environment E), instead, we acquired the temperature readings recorded every second at EPFL campus [1] from 210 sensors to simulate temperature variations over E. The extent of the area for E is different from that of EPFL campus and hence we had to spatially rescale the temperature readings. To this end, first since the sensors at EPFL campus are not uniformly located, we interpolated the temperature measurements at EPFL campus to find the temperature at uniformly distributed locations (we selected 600 locations). Afterward, we divided the space of E into 600 grid blocks with the block size of $1km \times 1km$ and assigned each EPFL temperature measurement to the corresponding grid block while preserving the locality of the readings. We denote the set of these grid blocks by G. The temperature variations within each grid block $g \in G$ is assumed to be negligible and therefore any point in g is a sufficiently accurate representation of the temperature of all the point in g.

5.1.3 Sensor Placement

We assume there is no direct communication between sensors and aggregators, and hence, the sensors and aggregators should be placed on R to leverage PDT for data transfer. Only a subset of grid blocks in G overlap with R, and therefore, as a filtering step we select a set of grid blocks $G' \subseteq G$ which overlap with G. As the temperature variations within a grid block is assumed to be negligible, a sensor placed anywhere within a grid block $g \in G'$ can capture the temperature for all the points in g. We adapt the greedy approach proposed in [9] for choosing the sensors locations. Given the set of initial grid block G', this algorithm chooses the next location for sensor placement which provides the maximum reduction in the entropy of predicting temperature at unsensed locations (locations where still no sensor is placed). More formally, assume the currently selected sensors locations is represented by S_C (initially $S_C = \phi$). At each iteration the algorithm selects a grid block $g \in G'$ and add its center g_c to S_C which maximizes

$$\left[H(S_C \cup g_c) - H(S_C \cup |\overline{S_C})\right] - \left[H(S_C) - H(S_C|\overline{S_C} \cup g_c)\right].$$

To simplify notation we use $\overline{S_C}$ to mean $V - (S_C \cup g_c)$. After $|S|$ iterations, the set S_C contains the sensor locations.

5.1.4 Aggregator Placement

Similar to sensors placement, we select a subset of grid blocks in G' to place the aggregators. We consider two different approaches to select aggregators locations. Next, we explain each approach in details.

RAND Approach: With the first approach, we randomly select $|A|$ grid blocks from G' where the set of selected grid blocks centers comprise the aggregators locations. We call this approach $RAND$.

FREQ Approach: With the second approach, we consider the frequency with which the vehicles visit the grid blocks in G'. Intuitively, the points which are visited more often over time are better candidates to place the aggregators. We denote this approach by $FREQ$. With FREQ, we select $|A|$ grid cells from G' which are visited the most. To find the frequency of visit for each grid cell, we used the vehicles positions recorded during the first 12 hours of the day. To prevent the aggregators to be placed very close to each other we consider a constraint such that the aggregators locations should be at least $d$km apart. The selected grid cells centers comprises the aggregators locations. We experimentally find the reasonable value for d in Section 5.2.1.

5.1.5 Routing Protocols

Variety of routing protocols are proposed to transmit a packet from a source to destination node in ad hoc/oppurtunitic networks. In this work, we study the performance of two of those protocols [16] in the context of PDT. We next explain each protocol in detail.

Epidemic Routing: The first protocol is *Epidemic* routing which floods the packets into the network. With epidemic routing, whenever two vehicles o_1 and

o_2 meet all the packets carried by o_1 (o_2) are transferred to o_2 (o_1). Clearly, with this approach packets are propagating fast between the vehicles but at the same time this approach consumes a huge amount of communication resources.

Random Routing: The second approach is *Random* routing. With random routing a transfer probability p_r is selected and a packet is transmitted with the probability of p_r at each contact. By changing the value of p_r one can control the amount of packet transfer in the network.

We selected these two algorithms as they are easy to implement and at the same time they enable us to change the amount of data transfer in the network and observe its consequences on the effectiveness of Q-PDT.

5.2 Results

In this section, we present the results of our experimental case study.

5.2.1 Aggregator Placement

In this experiment, we evaluate the efficiency of different aggregator placement approaches. We first study the efficiency of FREQ algorithm and afterward compare the effectiveness of FREQ and RAND approaches.

5.2.1.1 Efficiency of FREQ: With the FREQ approach, the selected aggregator locations should be at least $d$km apart. We varied d from 1km to 7km and measured the quality of data during a time interval $|I_i| = 1$ hour in order to study the impact of the selected d. Our results are shown in Figure 2 for $|S| = 60$ and $|A| = 15$. As the figure illustrates, there is a trade-off in selecting d. Increasing d results in selecting aggregator locations which are more widely distributed in the space, and hence, the aggregators are reachable from vehicles moving in various directions. On the other hand, by increasing d we are selecting points which are less visited by vehicles over time. Given our experimental settings, the best value for d is 3km for both routing protocols (see the figure).

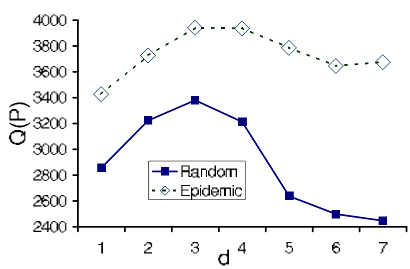
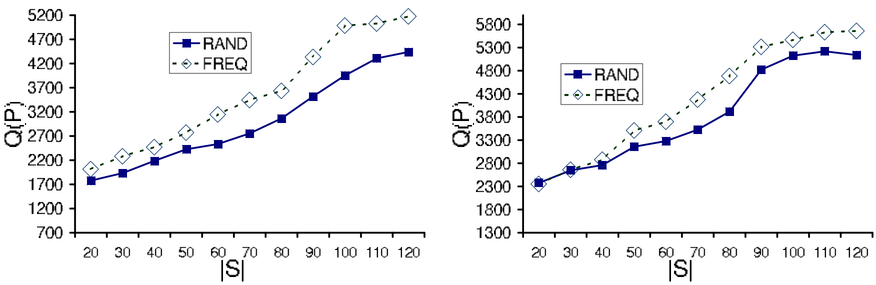


Fig. 2. Data quality vs. d

5.2.1.2 FREQ vs. RAND: Using the same settings as those of the experiments in the previous section, we compare the efficiency of the FREQ (d=3) and RAND placement approaches. Figure 3 shows the quality of data for different aggregators placement approaches and different routing protocols. The results show that the FREQ placement approach outperforms RAND. As expected, with FREQ, the quality of data is higher than RAND because aggregators are placed in the locations which are visited more often as compared to the of rest of the locations. Moreover, with RAND, in some cases aggregators are placed in locations which are not visited at all which makes the performance of this approach very poor. In the rest of the experiments in Section 5.2, we assume FREQ is used for aggregator placement.



(a) Random routing (b) Epidemic routing

Fig. 3. Data quality vs. $|S|$ for different aggregator placement approaches and routing protocols

5.2.2 The Count of Generated Packets

With this experiment, we study how the number of generated packets depends on the routing protocol and the number of sensors. The result is shown in Figure 4 for $p_r = 0.1$ and during a time interval $|I_i|$. As expected, as the number of sensors ($|S|$) increases, the number of generated packets grows. Moreover, the increase rate is much higher in two cases ($|S| = 20$ and $|S| = 100$). The reason is that for these two cases a set of sensors are placed at new locations in G' which are much more visited by cars than other locations and hence more packets are generated. On average over all the cases, the number of generated packets in epidemic routing is 90% more than random routing.

5.2.3 Data Quality Over Time

With this experiment, we evaluate the quality of the transferred data over time. To this end, we set $T = (I_1, I_2, \ldots .I_8)$ where $|I_i| = 1$ hour and T spans the time interval [12 20], $p_r = 0.1$ for random routing, $|S| = 60$ and $|A| = 15$. The result is shown in Figure 5. Over all the cases, epidemic routing is collecting data which has 49% higher quality than random routing. The difference between data quality for these two routing protocols varies between 41% and 64% where the highest difference occurs when the dynamism of the vehicles is greater, i.e., I_1=[12 13] and I_5=[16 17]. The reason is that when the dynamism of the vehicles are higher, more vehicles become reachable to each other, to sensors and to aggregators. With epidemic routing, the packets can directly hop through the reachable devices to travel from a sensor to an aggregator. However,

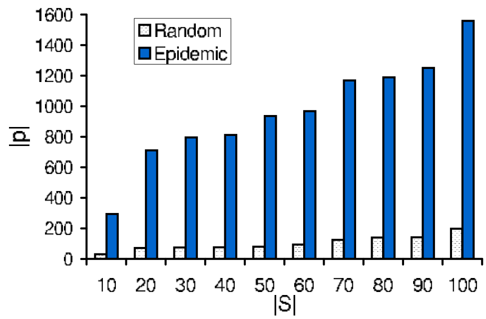


Fig. 4. Generated packets vs. number of placed sensors ($|S|$)

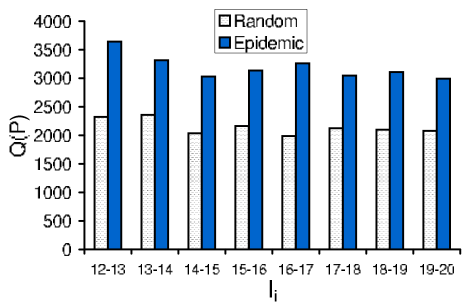


Fig. 5. Data quality vs. I_i

with random routing the probability that a packet p travels from one vehicle to another decreases as the number of vehicles which are traversed by p.

5.2.4 Effect of p_r

In this section, we study the effect of changing p_r for random routing. We set $|S| = 50$, $|A| = 13$, $T = I_1$ and varied p_r from 0.1 to 1 (with $p_r = 0$, no data transfer occurs and hence $Q(P) = 0$). Figure 6 shows the effect of changing p_r on $Q(P)$ for a time interval $|I_i| = 1$ hour. As expected, by increasing p_r, $Q(P)$ grows because there are more packet exchanges between vehicle and between vehicles and sensors/aggregators. The rate of increase for $Q(P)$ is generally decreasing when p_r grows and it is largest at $p_r = 0.2$.

5.3 Observations

We can summarize our observations based on the experimental case study as follows.

5.3.1 Sensor/Aggregtor Placement

The approach to place the aggregators which considers the dynamism of the vehicle (FREQ), outperforms the random aggregators placement approach in terms of the quality of transferred data (by 15% and 25% for random and

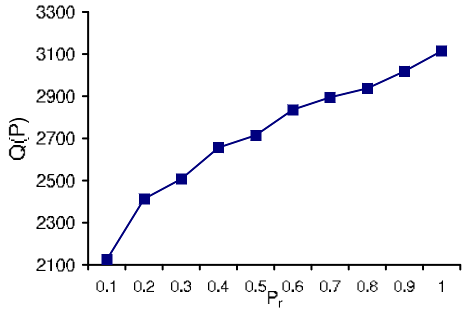


Fig. 6. Data quality vs. p_r

epidemic routing protocols, respectively). Moreover, the quality of the transferred data depends on the dynamism of the vehicles as well. The quality of the transferred data is higher when the dynamism of the vehicles is greater. With our experiments, we observed that 24% difference exists between quality of data for sub intervals in $T = [12 \quad 20]$. This suggests that an optimal solution for Q-PDT should consider the movement of the participating objects as well as the changes in their movement behavior over time.

5.3.2 Routing Protocols

The epidemic routing approach outperforms the random routing because of flooding the packets into the network of vehicles. However, the communication cost with epidemic routing is much higher. For example, with $p_r = 0.1$, the number of generated packets is 90% lower than that of epidemic routing where the data quality is lower by 49%. This shows that there is a trade-off in selecting p_r in terms of available communication resources and the desired sensing application quality. With our experiments, the difference between random and epidemic routing varies over time (see Section 5.2.3). Therefore, the value for p_r should change in optimal Q-PDT based on the movement behavior of the participating objects over time.

6 Related Work

PDT in the context of location-based social networks was proposed as a vision [15] to transfer packets for sensing systems which are deployed in the environments where no pre-existing communication infrastructure are available. In this work, we provided an experimental case study on using PDT in a more general setting in which a network of cars are participating in data transfer. With this case study, we studied various approaches for packets routing and sensors/aggregators placement by conducting experiments on real PDT data.

A body of relevant work is the literature on sensor deployment and sensing coverage in the field of sensor networks. In [6,11], the coverage problem is formulated as a decision problem to determine whether every point in the service area of the sensor network is covered by at least k sensors. However, with sensor deployment the goal is to maximize the coverage by proper sensor placement. Most proposed approaches for sensor deployment assume simple sensing models with circular (omnidirectional or unidirectional) coverage for sensors [5,17], whereas, these models are not optimal to model verity of phenomena. In [14] we introduced an approach for efficient placement of the visual sensors. Recently, in [8,9] Gaussian processes are used to model the monitored phenomena and consequently sensor placement. Moving sensors are studied recently to improve the coverage compared to static sensors [19].

The other category of related work is on opportunistic networking in mobile ad hoc networks. Mobile ad hoc networks are typically composed of mobile nodes that communicate over wireless links without any central control [2]. A variety of opportunistic routing algorithms are proposed to transfer a data packet from a source to a destination node in the network [13,10].

The difference between this work and the aforementioned two categories of related problems (i.e., sensor deployment/coverage and ad hoc opportunistic networking) is that we consider the objectives and constraints of both problems (which are partially competing constraints) concurrently.

7 Conclusions and Future Directions

In this paper, we introduced the problem of quality-aware participatory data transfer (Q-PDT). We presented an experimental case study on the problem based on real datasets including the GPS track data for more than 2000 vehicles in the city of Beijing. As part of our future work, we plan to study the Q-PDT problem on more datasets and also design optimal Q-PDT solutions for various applications based on the observations made in this work.

Acknowledgments. This research has been funded in part by NSF grant CNS-0831505 (CyberTrust), the NSF Integrated Media Systems Center (IMSC), unrestricted cash and equipment gift from Google, and in part from the METRANS Transportation Center, under grants from USDOT and Caltrans. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

References

1. `http://sensorscope.epfl.ch/`
2. Handbook of wireless networks and mobile computing. John Wiley & Sons, Inc., New York (2002)
3. U. Center for Embedded Networked Sensing (CENS), `http://urban.cens.ucla.edu/projects/`
4. Deshpande, A., Guestrin, C., Madden, S.R., Hellerstein, J.M., Hong, W.: Model-driven data acquisition in sensor networks. In: Proceedings of the Thirtieth International Conference on Very Large Data Bases, VLDB 2004, pp. 588–599. VLDB Endowment (2004)

5. Dhillon, S.S., Chakrabarty, K.: Sensor placement for effective coverage and surveillance in distributed sensor networks. In: Proc. of IEEE Wireless Communications and Networking Conference (2003)
6. Huang, C.-F., Tseng, Y.-C.: The coverage problem in a wireless sensor network. In: WSNA 2003, ACM, New York (2003)
7. Kashani, F.B., Shirani-Mehr, H., Pan, B., Bopp, N., Nocera, L., Shahabi, C.: Geosim: A geospatial data collection system for participatory urban texture documentation. IEEE Data Eng. Bull. (2010)
8. Krause, A., Rajagopal, R., Gupta, A., Guestrin, C.: Simultaneous placement and scheduling of sensors. In: IPSN 2009, Washington, DC, USA, IEEE Computer Society, Los Alamitos (2009)
9. Krause, A., Singh, A., Guestrin, C.: Near-optimal sensor placements in gaussian processes: Theory, efficient algorithms and empirical studies. J. Mach. Learn. Res. 9, 235–284 (2008)
10. Lindgren, A., Doria, A., Schelén, O.: Probabilistic routing in intermittently connected networks. SIGMOBILE Mob. Comput. Commun. Rev. (2003)
11. Meguerdichian, S., Koushanfar, F., Potkonjak, M., Srivastava, M.B.: Coverage problems in wireless ad-hoc sensor networks. In: IEEE INFOCOM (2001)
12. Mokkadem, A.: Estimation of the entropy and information of absolutely continuous random variables. IEEE Transactions on Information Theory 35(1), 193–196 (1989)
13. Ramanathan, R., Hansen, R., Basu, P., Rosales-Hain, R., Krishnan, R.: Prioritized epidemic routing for opportunistic networks. In: MobiOpp 2007, New York, NY, USA (2007)
14. Shirani-Mehr, H., Banaei-Kashani, F., Shahabi, C.: Efficient viewpoint selection for urban texture documentation. In: Trigoni, N., Markham, A., Nawaz, S. (eds.) GSN 2009. LNCS, vol. 5659, pp. 138–148. Springer, Heidelberg (2009)
15. Shirani-Mehr, H., Kashani, F.B., Shahabi, C.: Using location based social networks for quality-aware participatory data transfer. In: GIS-LBSN (2010)
16. Song, L., Kotz, D.F.: Evaluating opportunistic routing protocols with large realistic contact traces. In: Proceedings of the Second ACM Workshop on Challenged Networks, CHANTS 2007, pp. 35–42. ACM, New York (2007)
17. Wu, C.-H., Lee, K.-C., Chung, Y.-C.: A delaunay triangulation based method for wireless sensor network deployment. Comput. Commun. (2007)
18. Yin, J., ElBatt, T., Yeung, G., Ryu, B., Habermas, S., Krishnan, H., Talty, T.: Performance evaluation of safety applications over dsrc vehicular ad hoc networks. In: Proceedings of the 1st ACM International Workshop on Vehicular ad Hoc Networks, VANET 2004, pp. 1–9. ACM, New York (2004)
19. Yu, X., Zhao, H., Zhang, L., Wu, S., Krishnamachari, B., Li, V.O.K.: Cooperative sensing and compression in vehicular sensor networks for urban monitoring. In: IEEE International Conference on Communications (2010)

A Folksonomy-Based Recommendation System for the Sensor Web

Rohana Rezel and Steve Liang

Department of Geomatics Engineering, Schulich School of Engineering University of Calgary, 2500 University Drive NW, Calgary, Alberta T2N 1N4
{rdrezel,steve.liang}@ucalgary.ca
http://sensorweb.geomatics.ucalgary.ca/

Abstract. This paper introduces a folksonomy-based recommendation system for the worldwide Sensor Web which aims to aid users to deal with the terabytes of data that are generated by the sensors. We demonstrate how folksonomies could be adopted for the Sensor Web and other geospatial applications dealing with large volumes of data, by exploiting the geospatial information associated with three key components of such collaborative tagging systems: tags, resources and users. We propose algorithms for: i) suggesting tags for users during the tag input stage; ii) generating tag maps which provides for serendipitous browsing; and iii) personalized searching within the folksonomy. We experimentally evaluate our algorithms using an existing large dataset. An implementation of the folksonomy for the emerging Sensor Web platform is also presented.

1 Introduction

There are millions of sensors ranging from oceanic buoys to flood gauges, from traffic cameras to space-borne Earth observation platforms strewn around the globe to continuously collect information about our environment. The cost of deployment of these sensors is decreasing as advances are made in microelectronic and networking technologies and this is fuelling their further growth. The data produced by these networks are increasingly being disseminated via World Wide Web, leading to the formation of what is termed the worldwide Sensor Web[4][19].This process has been facilitated by the set of interoperability interfaces, named Sensor Web Enablement (SWE) put forward by the Open Geospatial Consortium (OGC). With SWE, OGC intends to provide a standard data infrastructure for the integration of these diverse sensors over the Internet[8].

The ability to monitor our planet via the Sensor Web presents opportunities for a multitude of stakeholders ranging from policy makers to researchers, but Raskino *et al.*[23] argue that the sheer volume of data produced *"will only be of value if the 'terabyte torrent' of data it generates can be collected, analyzed and interpreted."* Sequeda and Corcho[25] argue that it may be possible to achieve this by applying to sensor data *"the same principles that have been used for the publication of other types of (more static) data on the (Semantic) Web, in the context of the Linked Data initiative."*

K. Tanaka, P. Fröhlich, and K.-S. Kim (Eds.): W2GIS 2011, LNCS 6574, pp. 64–77, 2011.

The expansion of the Sensor Web's data volumes has been matched by the equally rapid growth of its user base. This has been partially spurred by the availability of user-friendly Sensor Web browsers, i.e., Virtual Globe-based browsers used to navigate these web enabled sensor networks. It is estimated that Google Earth's user base is more than 500 million users,[1] whilst NASA has claimed that its World Wind attracted 10 million users within the first year of operation growing at the rate of 100,000 new users per week.[2] Sensor web browser examples include the Microsoft Research SciScope,[3] Oak Ridge National Lab's Sensorpedia,[4] and University of Calgary's GeoCENS.[5]

The increased user participation opens up avenues to apply to the Sensor Web technologies used previously in the World Wide Web to tackle the Data Rich Information Poor (DRIP)[29] syndrome. Information retrieval, information filtering and machine learning were initially applied to the WWW to deal with the information overload but recent research has been more focused on automated collaborative filtering (ACF) coinciding with the emergence of Web 2.0. Maltz and Ehrlrich [20] define collaborative filtering as being *"based on the premise that people looking for information should be able to make use of what others have already found and evaluated."* It relies on the societal concept of "word of mouth", whereby users record their experience about a particular item of information they have encountered, usually in the form of annotations and ratings. ACF extends this by adding automation, scale, and anonymity[12].

Two main strategies for capturing user experiences in ACF are folksonomy and collaborative rating. The term "folksonomy", a combination of the words "folk" and "taxonomy" [31], is a lightweight system to share resources and the user created ad hoc labels associate with those resources. In taxonomy, the vocabularies are hierarchical and controlled, and experts are required to assign appropriate terms for the resources. In a folksonomy, the users are free to attach any term, called a tag in folksonomy nomenclature, to the resources in question. The control and design of a taxonomy is replaced in a folksonomy by the concepts of self-organization and emergence.

Web applications based on folksonomy such as Delicious,[6](for URL bookmarks) Flickr,[7](for images) and CiteULike[8](for academic publications) have grown exponentially over the last few years, both in terms of the number of users and the amount of information being shared. The has been largely attributed to the fact that users do not need to learn specific skills to use these tools[15]. The users have been able to create large volumes of information in a short time demonstrating that folksonomy-based approaches are able to overcome

[1] `http://earth.google.com/outreach/africa.html`
[2] `http://www.nasa.gov/centers/ames/news/releases/2006/06_30AR.html`
[3] `http://www.sciscope.org/`
[4] `http://www.sensorpedia.com/`
[5] `http://geocens.ca/`
[6] `http://delicious.com/`
[7] `http://www.flickr.com/`
[8] `http://www.citeulike.org/`

the "knowledge acquisition bottleneck"[14], which was a major limitation of knowledge-based systems of the past. .

By enabling knowledge discovery and information filtering, folksonomies could be used to tackle the previously described challenge of handling the "terabyte torrent" of data that the Sensor Web will face in the future.

In this paper, we introduce a folksonomy-based recommendation system for the Sensor Web by extending collaborative tagging techniques to geospatial contexts. Whilst the methods described in this paper were developed for the Sensor Web, they are equally applicable to other geospatial applications. However, discussion of such general cases are beyond the scope of this paper.

Much of the preliminary research related to this task was reported in our earlier paper[24], and in this research we introduce an improved set of algorithms and present experimental results.

1.1 Contribution

Wang *et al.* [32] have identified three common user tasks in a folksonomy-system divided into to two phases, and our research extends each of these tasks in to the geospatial domain in order for them to be applicable to the Sensor Web.

Indexing Phase

1. Collaborative tagging: the task of a user assigning a tag to a resource such as an URL (e.g., delicious.com), a picture (e.g., Flickr) or an article (e.g., CiteULike). Users assigning these tags in a folksonomy are not necessarily experts and may not be aware of the tag vocabulary that has already been constructed by other users. For example, *NY*, *New York* and *Newyork* could all be variations of the same tag. One way to assist the users in this regard and ensure some consistency in tagging is by making personalized tag suggestions for the resources they are trying to assign tags to. In this paper we present an algorithm to suggest tags by exploiting the geospatial attributes of the resources and tags on the Sensor Web into consideration.

Information Retrieval Phase

2. Collaborative browsing: task of a user navigating through the tags collected in the system which could potentially lead to serendipitous discovery of resources. Tag clouds, where the size of each tag corresponds to some measure of its prominence, have been traditionally used for browsing folksonomies. Their counter part in geospatial folksonomies is the tag map, where the location of each tag in the tag cloud is dependent on some geospatial characteristic of that tag. We present a new algorithm for generating tag maps for Sensor Web browsers where the geospatial attributes of each tag assignment is taken into account.

3. Collaborative Searching: task of a user retrieving resources based on tag queries. This is accomplished either by clicking through a tag on the tag map, or by typing a tag into a search form. This research describes an algorithm for enhanced query processing by utilizing the geospatial aspects of the queries and data.

2 Related Work

2.1 Semantic Sensor Web

Semantic Sensor Web(SSW) has been proposed as means of alleviating the "too much data and not enough knowledge" problem mentioned above[26]. SSW adopts an ontology-based approach building on top of the W3C Semantic Web envisioned by World Wide Web inventor Tim Berners-Lee[7]. The drawback of the SSW approach is that, as Xu *et al.* [34] describes, *"A universal ontology is difficult and expensive to construct and maintain when there involve hundreds of millions of users with diverse background."*

2.2 Tag Suggestion

An early technique proposed for tag suggestion took into consideration multiple criteria such as coverage of multiple facets, least effort, and high popularity[34]. The recommendation algorithm proposed by Benz et al.[6] is based on the number of users, the similarity of users, the frequency of a tag and the specificity of the tag. Lossless compression over existing tag data is used in the TagAssist[28] approach to improved the quality of suggested tags. Heymann *et al.*[13] use association rule mining to suggest tags. Krestel *et al.*[17] use Latent Dirichlet Allocation (LDA), whereby resources with a fairly stable and complete tag set are used to suggest tags for a new resource through the mapping of common topics.

2.3 Tag Browsing

The use of tag maps to to visualize large collections of georeferenced photos was pioneered by Yahoo! Research Berkeley. [16] [3]. Tag maps have been further enhanced to convey additional information through the use of various symbolisms such as different colours for tags[33].

Tag map research up to now has been focused on folksonomies where the resources are photographs, which are point data. However, as a resource in Sensor Web applications can be points, lines or regions, tag maps used for Sensor Web visualization will have to take this case into consideration.

2.4 Tag Searching

FolkRank[14] algorithm, which is inspired by Google's seminal PageRank[9] algorithm, is one of the earliest algorithms developed for searching folksonomies. The similarity estimation algorithm SocialSimRank[5] seeks to bridge the matching gap that exists between the annotations and queries (e.g., "shop" and "shopping"). GroupMe![2] is a folksonomy application that allows users to group Web resources of interest, and search strategies developed for GroupMe! take contextual information about the resources into consideration during search. Resource retrieval is improved in Triple Play[1] by smoothing the tag space by using tag-resource and user-tag relationship information found int folksonomies. TC-Social Rank[11] attempts to improve information retrieval in folksonomy search algorithms by exploiting temporal information and user click statistics found in tagging systems.

3 Folksonomies in Geospatial Contexts

According Hotho *et al.*[14], the formal definition of a folksonomy $\mathbb{F}$ is

$$\mathbb{F} = (U, T, R, Y) \quad (1)$$

where U,T,R are the finite sets of users, tags and resources respectively, and Y is the ternary relationship between U,T and R, called the tag assignment where

$Y \subseteq U \times T \times R$

Extending this to a geospatial context where each item only has one geographical location, and a user can only tag one item once, we add the function

$\phi : (U, T, R) \to p$

where p is the geographical location (latitude, longitude) where the tag is attached to a resource in the system.

However, in a geospatial context where users can tag a single item multiple times at multiple locations. Hence we extend the function as follows:

$\varphi : (U, T, R) \to P$

where P is the set of locations.

This is an important consideration for Sensor Web, where a resource such as a Web Map Server (WMS) map can be considered as a sensor web server serving remote sensing data. And a WMS map can cover a large area, and the users may tag this map with the same tag at multiple locations.

Furthermore the tag assignment Y has to be extended to be:

$\mathbb{Y} \subseteq U \times T \times R \times P$

Thus, the geospatial folksonomy $\mathbb{F}_{geo}$ can be expressed as the pentuple

$$\mathbb{F}_{geo} = (U, T, R, \mathbb{Y}, \varphi) \quad (2)$$

3.1 Tag Suggestions

In a non-geospatial folksonomy system, when a user $u \in U$, is preparing to tag resource $r \in R$, the recommended tags are ranked according to the weighted frequency of the tag $t \in T$, for the resource$r$, using the equation

$$Freq\,(r,t,u) = \sum_{v \in neigh(u)} Sim\,(u,v)\,|\{(v,t,r) \in Y\}| \tag{3}$$

where function $Sim\,(u,v)$ is the similarity function between user u and their neighbor v, determined by the neighborhood function $neigh(u)$. The strategies for the computation of similarity are discussed in Section 3.3.

The term $|\{(v,t,r) \in Y\}|$ evaluates to either 1 or 0 in a folksonomy, depending on whether the tag assignment exists or does not exist respectively. Thus, $\sum_{v \in neigh(u)} |\{(v,t,r) \in Y\}|$ returns the count of all instances where users u and v have tagged the same resource. The term $Sim\,(u,v)$ weights each occurrence according to the similarity between the users.

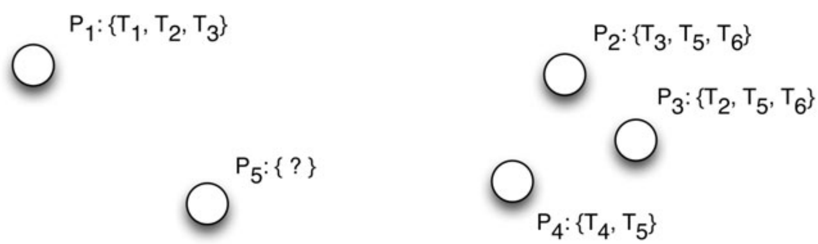


Fig. 1. Example Tagging Scenario

However, take the scenario in 1, where we assume for the sake of simplicity that there is only one resource, and one user. Tags $T_1 \ldots T_6$ are placed at points $P_1 \ldots P_4$, in the manner shown in the figure, and the user is about to place a tag at point P_5. If only the frequency is calculated, that tag T_5 would be the first suggestion. However, Tobler's first law of geography[30] states that:

> *"Everything is related to everything else, but near things are more related than distant things."*

Therefore, the effect of P_1 being closer to P_5 has to be taken into account when recommending tags for P_5. Thus, equation 3 is modified as follows:

$$Freq\,(r,t,u,p) = \sum_{v \in neigh(u)} \sum_{q \in \varphi(v,t,r)} Sim\,(u,v)\,\frac{|\{(v,t,r,q) \in \mathbb{Y}\}|}{Dist\,(p,q)} \tag{4}$$

where

- p is the geographical point at which tag t is inserted on this instance
- function $Dist\,(p,q)$ is the distance function between points p and q

Thus, equation 4 is the inverse distance weighted form of equation 3.

3.2 Tag Browsing

The naïve non-geospatial method to generate tag clouds relies on calculating the weighted frequency of each tag t as

$$Freq\,(t,u) = \sum_{r \in R} \sum_{v \in neigh(u)} Sim\,(u,v)\,|\{(v,t,r) \in Y\}| \tag{5}$$

In visualization, the font size with which tag t is drawn corresponds to the $Freq$ score.

Extending 5 to geospatial contexts requires two modifications:

Modification 1: Calculate the $Freq$ score taking into consideration the distance from the user eye-point (the center of the users display) a.

Modification 2: Place the tags on the users display in a meaningful and a visually pleasing manner.

For the first modification, equation 5 is extended as:

$$Freq\,(t,u) = \sum_{r \in R} \sum_{v \in neigh(u)} \sum_{q \in \varphi(v,t,r)} Sim\,(u,v)\,\frac{|\{(v,t,r,q) \in \mathbb{Y}\}|}{Dist\,(a,q)} \tag{6}$$

where

- q is the point at which user v tagged resource r with tag t
- a is the user's current eye-point
- $Dist(a,q)$ is the distance function between points p and a

For the second modification, algorithm 1 is used to calculate the tag positions:

foreach $t \in T$ **do**
  Calculate centroid c of the set of all the points $j \in u \in, Ur \in R, \varphi\,(u,t,r)$
  Calculate distance d_{ac} from eye-point a to c
end
Calculate d_{max}, the maximum of all d_{ac}
foreach $t \in T$ **do**
  Calculate d_{ab}, the distance from point a to the point b where line ac intercepts the viewport Calculate $d_{ag} = \frac{d_{ac}}{d_{max}} d_{ab}$ Display tag t at g, where g is a point on line ab, and length of $ag = d_{ag}$
end

Algorithm 1. Tagmap Calculation Algorithm

3.3 Tag Searching

The naïve non-geospatial method to calculate the ranking of a resource r for a given search tag t depends on the frequency of r being associated with t, and can once again be calculated using equation 3. This can be formally defined by using equation 6, with the modification that a is the user's eye-point at the point in time when the user initiates the search query.

3.4 User Similarity

The naïve method to calculate the similarity between the users u and v is to calculate the frequency with which the users have tagged the same resource with the same tag as:

$$Sim\left(u,v\right)=\sum_{t\in T}\sum_{r\in R}\left|\left\{(u,t,r)\in Y,(v,t,r)\in Y\right\}\right| \tag{7}$$

Equation 7 states that the similarity of users u and v is the count of times users u and v tagged the same resource with the same tag.

According to Tobler's law, however, users placing tags geographically close to other users' tags should have an effect on this similarity. Moreover, Matyas and Schlieder[21] make the assumption that:

> *"People who agreed on the qualities of one geographic region are likely to agree on the qualities of other geographic regions too."*

The above assumption is suitable where users are able to use ratings. However, in folksonomies, it is difficult to objectively measure the agreement on qualities. Thus, we relax the assumption as follows:

> *"People who have agreed on the tags for a given resource in one geographic region are likely to agree on the tags for the same resource to be placed at another geographic region too."*

Hence, we modify equation 7 to be:

$$Sim\left(u,v\right)=\sum_{t\in T}\sum_{r\in R}\sum_{p\in\varphi(u,t,r)}\sum_{q\in\varphi(v,t,r)}\frac{\left|\left\{(u,t,r,p)\in\mathbb{Y},(v,t,r,q)\in\mathbb{Y}\right\}\right|}{Dist\left(p,q\right)} \tag{8}$$

Thus, equation 8 can be described as the inverse distance weighted version of equation 8.

However a limitation of equation 8 is that users will have higher similarity scores with neighbours who are more prolific taggers. To remove this bias, we reduce the influence of such users by modifying equation 8 to be:

$$Sim\left(u,v\right)=\sum_{t\in T}\sum_{r\in R}\sum_{p\in\varphi(u,t,r)}\sum_{q\in\varphi(v,t,r)}\frac{\left|\left\{(u,t,r,p)\in\mathbb{Y},(v,t,r,q)\in\mathbb{Y}\right\}\right|}{n\cdot Dist\left(p,q\right)} \tag{9}$$

Where $n=\sum_{t\in T}\sum_{r\in R}\sum_{q\in\varphi(v,t,r)}\left|\left\{(v,t,r,q)\in\mathbb{Y}\right\}\right|$,*i.e.*,the total number of tagging instances for the user v in the folksonomy.

4 Implementation

The geospatial folksonomy described above is actively used for GeoCENS, a suite of applications for the Sensor Web currently comprising:

- A Sensor Web browser which is capable of retrieving and presenting information obtained from information sources using an array of Open Geospatial Consortium(OGC) standards such as Sensor Observation Service (SOS)[22], Sensor Planning Service (SPS)[27], Web Map Service (WMS)[18] and Web Coverage Service (WCS)[10]. The GeoCENS browser visualizes the data on top of NASA's virtual globe World Wind[9].
- An implementation of an OGC SOS server.

Figure 2 is a screen capture of the tag suggestion technique used for in generating terms for an autocomplete widget, while Figure 3 shows the tag map in play.

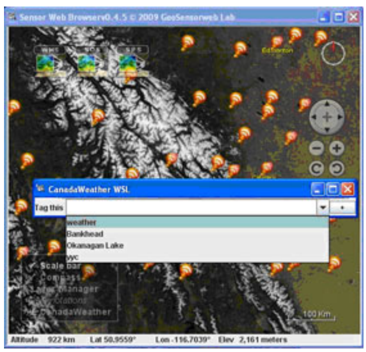


Fig. 2. GeoCENS Tag Suggestion

5 Experimental Evaluation

We preferred to test the robustness of our algorithms by using a large and well-established folksonomy rather than using the folksonomy of the GeoCENS platform, which is currently in beta testing phase and so far has not captured a sufficient volume of users generated data. Therefore we decided to use the folksonomy dataset from the popular web application Flickr as the surrogate. We are using the Flickr georeferencd photos as our data source. This data, obtained via the Flickr API, is a very large data set that can mimic a GeoCENS folksonomy for evaluation purposes. In section 5.1, we discuss how we mapped Flickr data to the GeoCENS data requirements.

It should be noted here that we are not intending to carry out an analysis of the Flickr data. Instead we are borrowing its folksonomy to test the validity of our proposed algorithms.

[9] http://worldwind.arc.nasa.gov/

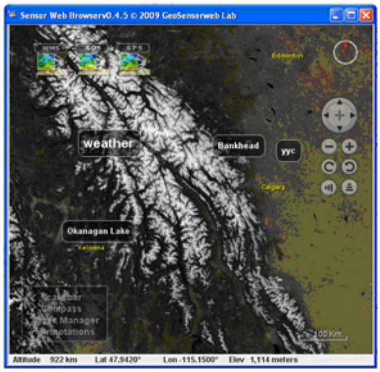


Fig. 3. GeoCENS Tag Map

5.1 Experimental Setup

In the GeoCENS implementation, a resource is a Sensor Web layer.[10] A Sensor Web layer example can be a collection of air temperature sensors grouped as a GIS layer. In the experiment, we use a Flickr photoset to mimic a Sensor Web layer, and we use each photo within the photoset to mimic a sensor node of the layer.Thus, as with GeoCENS, this represents a geospatial context where users can tag a single resource multiple times at multiple locations.

The Flickr dataset was most suitable for testing our tag suggestion algorithm. Therefore, we limited our experiment to that task. Table 1 describes the vital statistics of our data set. The scoring was done using the following algorithm:

Table 1. Experimental Data Set

Component	Sensor Web Equivlalent	Flickr Equivalent	Number of Items
Tag Assignments $\mathbb{Y}$	Tags Assignment	Tags Assignment	101,216
Users U	Users	Users	632
Tags T	Tags	Tags	17,566
Resources R	Sensor Web layer	Photosets	434
Points P	Sensor nodes	Photos	3,141

The algorithm aims to predict what the user would assign to a given resource using the training dataset. Then, it checks to see if the tag given to the resource by the user includes the predicted tag, and at what position.

The distance metric used is the great circle distance between points. We used d^k as the distance function, and we carried out the simulation for different values of k.

[10] To be more specific, a GeoCENS Sensor Web layer is in fact an OGC SOS server's Observation Offering. Details of OGC SOS definitions and terminologies can be found at the URL `http://www.opengeospatial.org/standards/sos`

Set score=0
Divided the extracted data into two sets: training set S and the evaluation set E
foreach *point* γ *in* E **do**
    foreach *resource* ρ *in* S **do**
        foreach *user* υ *in* S **do**
            Calculated the ordered list of suggested tags $\mathbb{T}$ using the candidate algorithm from data in S
            foreach *user* υ *has used tag* $\tau \in \mathbb{T}$ **do**
                Add $(|\mathbb{T}| - i)$ to score, where i is rank τ within $\mathbb{T}$
            end
        end
    end
end

Algorithm 2. Algorithm for calculating the Averaged Ranked Score

5.2 Experimental Results

Table 2 shows the results of our experiment.

Table 2. Results for different values of k

Values of k in d^k	Hit Rate (%)	Averaged Rank Score
1	28.99	4.67
2	26.66	4.42
3	25.73	4.22

The hit rate is percentage of times the algorithm correctly predicts the tag attached to a point by a user.

5.3 Limatations of the Experimental Setup

One major drawback of using Flickr data is that a point (a photo) cannot have the same tag assigned by two different users. Thus, sometimes, the user is unable to use the tags predicted by our algorithm because another user has tagged that photo with the same tag. The result of this is that our hit rate is lowered significantly.

6 Conclusion and Further Work

6.1 Conclusion

In this paper, we have illustrated how folksonomies can be extended to the Sensor Web. We presented how each of the key tasks of a folksonomy system, *i.e.*, tagging, browsing and searching, can be extended to provide support for geospatial applications. We have presented an implementation of these algorithms in the GeoCENS Sensor Web, a real-world Sensor Web implementation. We have done an experimental evaluation using the Flickr georeferenced photo dataset to mimic sensor web data.

6.2 Future Work

As mentioned earlier in the article, more research is needed into placement of tags on the tag map. How the location of tag placement impacts user similarity also needs to investigated. The methods discussed here could be further enhanced by taking temporal information into consideration.

Moreover, the automated collaborative filtering capabilities of the system can be further improved by introducing a rating system to complement the folksonomy system. With ratings, users will be able to record their like or dislike of a piece of information. This information can be used to calculate user similarities and recommend items to users. Furthermore, the ratings, although subject to bias, will give other users an indication about the quality or utility of an item.

Acknowledgments. This research has been made possible by the funding and support of the GeoCENS partners: CANARIE, Cybera, Biogeoscience Institute of the University of Calgary, Open Geospatial Consortium (OGC)(R) and the IP3 Research Network.

References

1. Abbasi, R., Staab, S.: Introducing triple play for improved resource retrieval in collaborative tagging systems. In: Proceedings of Exploiting Semantic Annotations in Information Retrieval (ESAIR), Workshop at ECIR 2008 (2008)
2. Abel, F., Frank, M., Henze, N., Krause, D., Plappert, D., Siehndel, P.: GroupMe!-Where semantic web meets web 2.0. In: Aberer, K., Choi, K.-S., Noy, N., Allemang, D., Lee, K.-I., Nixon, L.J.B., Golbeck, J., Mika, P., Maynard, D., Mizoguchi, R., Schreiber, G., Cudré-Mauroux, P. (eds.) ASWC 2007 and ISWC 2007. LNCS, vol. 4825, p. 871. Springer, Heidelberg (2007)
3. Ahern, S., Naaman, M., Nair, R., Yang, J.H.I.: World explorer: visualizing aggregate data from unstructured text in geo-referenced collections. In: Proceedings of the 7th ACM/IEEE-CS Joint Conference on Digital Libraries, JCDL 2007, pp. 1–10. ACM, New York (2007), http://doi.acm.org/10.1145/1255175.1255177
4. Balazinska, M., Deshpande, A., Franklin, M.J., Gibbons, P.B., Gray, J., Hansen, M., Liebhold, M., Nath, S., Szalay, A., Tao, V.: Data management in the worldwide sensor web. IEEE Pervasive Computing 6(2), 30–40 (2007)
5. Bao, S., Xue, G., Wu, X., Yu, Y., Fei, B., Su, Z.: Optimizing web search using social annotations. In: Proceedings of the 16th International Conference on World Wide Web, pp. 501–510. ACM, Banff (2007), http://portal.acm.org/citation.cfm?id=1242640&dl=GUIDE&coll=GUIDE&CFID=66663630&CFTOKEN=61226264
6. Benz, D., Tso, K., Schmidt-Thieme, L.: Automatic bookmark classification: A collaborative approach. In: Proceedings of the Second Workshop on Innovations in Web Infrastructure (IWI 2006), Edinburgh, Scotland (2006)

7. Berners-Lee, T., Hendler, J.: Scientific publishing on the semantic web. Nature 410, 1023–1024 (2001)
8. Botts, M., Percivall, G., Reed, C., Davidson, J.: OGC (R) sensor web enablement: Overview and high level architecture. In: Nittel, S., Labrinidis, A., Stefanidis, A. (eds.) GSN 2006. LNCS, vol. 4540, pp. 175–190. Springer, Heidelberg (2008)
9. Brin, S., Page, L., Motwami, R., Winograd, T.: The PageRank citation ranking: bringing order to the web. In: Proceedings of ASIS 1998, pp. 161–172 (1998)
10. Evans, J.D.: Web coverage service (WCS). OpenGIS Implementation Specification. Open Geospatial Consortium (2003)
11. Gulli, A., Cataudella, S., Foschini, L.: TC-SocialRank: Ranking the social web. Algorithms and Models for the Web-Graph, 143–154 (2009), http://dx.doi.org/10.1007/978-3-540-95995-3_12
12. Herlocker, J.L.: Understanding and Improving Automated Collaborative Filtering Systems, Ph.D. thesis, Phd Thesis. Department of Computer Science and Engineering, University of Minnesota, Minneapolis-MN-USA (2000)
13. Heymann, P., Ramage, D., Garcia-Molina, H.: Social tag prediction. In: Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2008, pp. 531–538. ACM, New York (2008), http://doi.acm.org/10.1145/1390334.1390425
14. Hotho, A., Jaschke, R., Schmitz, C., Stumme, G.: Information retrieval in folksonomies: Search and ranking. In: Sure, Y., Domingue, J. (eds.) ESWC 2006. LNCS, vol. 4011, p. 411. Springer, Heidelberg (2006)
15. Hotho, A., Jschke, R., Schmitz, C., Stumme, G.: Trend detection in folksonomies. In: Avrithis, Y., Kompatsiaris, Y., Staab, S., O'Connor, N.E. (eds.) SAMT 2006. LNCS, vol. 4306, pp. 56–70. Springer, Heidelberg (2006), http://www.kde.cs.uni-kassel.de/stumme/papers/2006/hotho2006trend.pdf
16. Jaffe, A., Naaman, M., Tassa, T., Davis, M.: Generating summaries and visualization for large collections of geo-referenced photographs. In: Proceedings of the 8th ACM International Workshop on Multimedia Information Retrieval, p. 98 (2006)
17. Krestel, R., Fankhauser, P., Nejdl, W.: Latent dirichlet allocation for tag recommendation. In: Proceedings of the Third ACM Conference on Recommender Systems, pp. 61–68. ACM, New York (2009), http://portal.acm.org/citation.cfm?id=1639714.1639726
18. de La Beaujardire, J.: Web map service implementation specification. Open GIS Consortium 82 (2002)
19. Liang, S.H., Croitoru, A., Tao, C.V.: A distributed geospatial infrastructure for sensor web. Computers & Geosciences 31(2), 221–231 (2005), http://www.sciencedirect.com/science/article/B6V7D-4F29HNY-1/2/afd0e9c8fd11bb78647049f232f2d7c5 Geospatial Research in Europe: AGILE 2003
20. Maltz, D., Ehrlich, K.: Pointing the way: Active collaborative filtering. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, pp. 202–209 (1995)
21. Matyas, C., Schlieder, C.: A spatial user similarity measure for geographic recommender systems. In: GeoSpatial Semantics: Third International Conference, GeoS 2009, Proceedings, Mexico City, Mexico, December 3-4 (2009)
22. Na, A., Priest, M.: Sensor observation service. Open Geospatial Consortium Inc., OGC (2006)
23. Raskino, M., Fenn, J., Alexander, L.: Extracting value from the massively connected world of 2015. Tech. Rep. G00125949 (April 2005), http://www.gartner.com/resources/125900/125949/extracting_valu.pdf

24. Rezel, R., Liang, S.: SWE-FE: extending folksonomies to the sensor web. In: 2010 International Symposium on Collaborative Technologies and Systems (CTS), pp. 349–356 (2010)
25. Sequeda, J., Corcho, O.: Linked stream data: A position paper, pp. 148–157 (October 2009)
26. Sheth, A., Henson, C., Sahoo, S.S.: Semantic sensor web. IEEE Internet Computing, 78–83 (2008)
27. Simonis, I.: OpenGIS sensor planning service implementation specification (2005)
28. Sood, S., Owsley, S., Hammond, K., Birnbaum, L.: Tagassist: Automatic tag suggestion for blog posts. In: Proceedings of the International Conference on Weblogs and Social Media, ICWSM 2007 (2007)
29. Tien, J.M.: Automated correctional data systems. Comp. Environ. Urban Syst. 10(3), 157–164 (1986)
30. Tobler, W.: A computer model simulating urban growth in the detroit region. Economic Geography 46(2), 234–240 (1970)
31. Wal, T.V.: Folksonomy:: vanderwal.net (February 2007), http://www.vanderwal.net/folksonomy.html
32. Wang, J., Clements, M., Yang, J., de Vries, A.P., Reinders, M.J.: Personalization of tagging systems. Information Processing & Management 46(1), 58–70 (2010), http://www.sciencedirect.com/science/article/B6VC8-4WWFN9R-1/2/33b9dc8198dc2220202cd1342f51fbfa
33. Wood, J., Dykes, J., Slingsby, A., Clarke, K.: Interactive visual exploration of a large spatio-temporal dataset: Reflections on a geovisualization mashup. IEEE Transactions on Visualization and Computer Graphics 13(6), 1176 (2007)
34. Xu, Z., Fu, Y., Mao, J., Su, D.: Towards the semantic web: Collaborative tag suggestions. In: Collaborative Web Tagging Workshop at WWW 2006, Edinburgh, Scotland (2006)

Semantically Enriching VGI in Support of Implicit Feedback Analysis

Andrea Ballatore and Michela Bertolotto

School of Computer Science and Informatics,
University College Dublin, Belfield, Dublin 4, Ireland
{andrea.ballatore,michela.bertolotto}@ucd.ie

Abstract. In recent years, the proliferation of Volunteered Geographic Information (VGI) has enabled many Internet users to contribute to the construction of rich and increasingly complex spatial datasets. This growth of geo-referenced information and the often loose semantic structure of such data have resulted in spatial information overload. For this reason, a semantic gap has emerged between unstructured geo-spatial datasets and high-level ontological concepts. Filling this semantic gap can help reduce spatial information overload, therefore facilitating both user interactions and the analysis of such interaction. Implicit Feedback analysis is the focus of our work. In this paper we address this problem by proposing a system that executes spatial discovery queries. Our system combines a semantically-rich and spatially-poor ontology (DBpedia) with a spatially-rich and semantically-poor VGI dataset (OpenStreetMap). This technique differs from existing ones, such as the aggregated dataset LinkedGeoData, as it is focused on user interest analysis and takes map scale into account. System architecture, functionality and preliminary results gathered about the system performance are discussed.

Keywords: Geographic Information Systems, Geo-Ontologies, DBpedia, LinkedGeoData, OpenStreetMap, Volunteered Geographic Information, Implicit Feedback Analysis.

1 Introduction

In recent years several spatial datasets have been published and have become available through open-source licenses. Crowd sourcing has played a major role in this process, enabling many Internet users to collaborate in the construction of rich and increasingly complex units of reusable geographical knowledge. As Volunteered Geographic Information (VGI) proliferates, the need for reduction of information overload and for semantic structure has become prominent. Since the late 90s, the lack of semantic structure on the Internet has become problematic for several applications and has promoted several initiatives in the context of the so-called *Semantic Web* [3]. This problem is very evident in the spatial domain and even more in VGI repositories where finding a meaningful structure of associations between geographic entities is far from straightforward.

K. Tanaka, P. Fröhlich, and K.-S. Kim (Eds.): W2GIS 2011, LNCS 6574, pp. 78–93, 2011.

We believe that the semantic gap between rather unstructured geographical data and higher level spatial and non-spatial concepts is one of the key challenges for reducing information overload in modern Geographical Information Systems. Our previous work has tried to address spatial information overload by generating personalised maps that match user interests and facilitate their tasks [20]. In this context, filling the semantic gap would help identify relevant information and generating meaningful user profiles.

Geo-ontologies have proven useful in the attempt to semantically enrich spatial datasets. Although existing VGI projects offer rich and complex datasets, because of the semantic gap, it is not easy to combine them with such geo-ontologies in the reduction of information overload. For this reason, we propose an approach for exploiting ontologies to perform spatial exploratory queries in the context of spatial knowledge and data discovery. Within this context, we have developed a system that integrates a spatially rich but semantically poor vector dataset (OpenStreetMap) with a spatially poor but semantically rich ontology (DBpedia), partially exploiting the mapping offered by an aggregated dataset (LinkedGeoData). When the user clicks on a geo-location on the map, the system processes a spatial query extracting spatial features from the vector dataset. These features are then mapped to ontological nodes, showing semantic contents to the user.

This system was developed in support of our implicit user profiling efforts that rely on the analysis of mouse movements on an interactive map to gather information about the user spatial interests. Compared to existing projects such as LinkedGeoData, our system is specifically designed to enhance semantic content in the field of implicit feedback analysis. In order to emphasise the user perspective in the interaction with spatial data, the map scale is taken into account during the extraction of ontological concepts, selecting features that are visible on the map at a given scale.

This is a new contribution as to the best of our knowledge no other similar project takes this aspect into account. Therefore we aim at overcoming this limitation for our purposes.

Indeed, when a user interacts with a Web map, the extended semantic knowledge extracted by such a system can be used to further refine the insight into the user's spatial interests. In this paper we show that our approach to extracting explicit semantic relationships between vector geographic data and ontological entities can benefit spatial user profiling through unobtrusive implicit feedback indicators, such as mouse movements.

The remainder of this paper is organised as follows: Section 2 discusses related work in the area of geo-spatial ontologies, volunteered geographic information and geographical crowdsourced information. Section 3 outlines the system architecture including a detailed description of the technologies and web services used. Section 4 describes the system functionality, while Section 5 details the system Web graphical user interface and describes a preliminary evaluation. Finally, Section 6 draws a conclusion, identifying limitations and directions for future work.

2 Related Work

Finding meaning in unstructured - or loosely structured - online information has been the main challenged addressed by the set of initiatives under the name *Semantic Web* at the beginning of the millennium [3]. After a decade, ontologies are still considered one of the key elements of the Semantic Web and are technologically supported by languages such as the Resource Description Framework (RDF) and the Web Ontology Language (OWL). Given that there is no universally accepted definition, in this work we indicate as an *ontology* a document that formally defines the relations among terms. Ding et al. provide a detailed survey of the field of ontologies for the Semantic Web [7].

Personalisation is one of the fields in which ontologies have been utilised to build user profiles in location-based services [27,12], to develop context-aware mobile systems [17] and to refine Web searches through ontological user profiles [23]. Although ontologies have a long and well-established tradition, little work has been done in the area of spatial implicit feedback analysis.

The last decade has witnessed another major Internet phenomenon, sometimes called *neogeography* in the literature [25]. During this transition, the traditional GIS field has progressively become Web-based and universally accessible through Web technologies, a process described by Haklay et al. [13]. Once collaborative tools reached a certain maturity, online efforts have started, resulting in several high-impact GIS-related projects, following the crowdsourcing phenomenon. The mainly non-spatial project Wikipedia still represents the most visible product of these processes. This transformation of users from passive end points of GIS geographical services to active contributors has been defined *Volunteered Geographic Information* [11].

As the importance of crowdsourcing and Web GIS grows steadily, the semantic gap has become increasingly problematic and started attracting further initiatives. Spatial ontologies have traditionally helped geographical knowledge to be efficiently stored, processed and shared among institutions. Fonseca et al. have conducted influential work in this area [9,10]. Several projects have sprung in the area of crowdsourcing and specifically in VGI. In this work we chose OpenStreetMap[1] as the data repository for bottom-up user-generated geographical vector data. Being one of the foremost VGI projects, OpenStreetMap has attracted a lot of contributors and researchers [14]. The quality of the OpenStreetMap spatial data and the VGI production mode are object of controversy, similarly to Wiki debate [8]. The project constitutes a typical example of a vast and fast-growing dataset built by users on an limited underlying semantic structure [4, p.383]. OpenStreetMap does not impose a formal semantic structure but rather suggests a ‘core recommended feature set and corresponding tags’[2].

In the non-spatial domain, Wikipedia[3] is indeed the most impressive user-generated dataset available. As Völkel et al. pointed out, Wikipedia pages are

[1] http://www.openstreetmap.org
[2] http://wiki.openstreetmap.org/wiki/Map_Features
[3] http://www.wikipedia.org

very useful but they are not easy to be machine-processed [26]. Diverse efforts have been made to bridge this semantic gap over the past few years and progressively converged into DBpedia[4], a unified dataset offering a semantically structured version of Wikipedia [1,6]. In order to structure and classify the pages, DBpedia relies on manually created cross-domain ontology, based on the most commonly used infoboxes within Wikipedia. From a spatial viewpoint DBpedia contains geo-coordinates for 390,000+ geographic locations but not the complex polygons and polylines that are available in OpenStreetMap.

Within this context, LinkedGeoData[5] aims to the challenging task of integrating such OpenStreetMap geometries with DBpedia and other semantically rich datasets [2]. Similarly to DBpedia, LinkedGeoData contains an ontology extracted from OpenStreetMap. The main dataset of the project contains about 3 billion RDF entities, creating a ontologically enhanced version of OpenStreetMap. For distributing its datasets, LinkedGeoData follows the 4 rules of the Linked Data initiative[6], whose increasing impact is analysed by Bizer et al. [5]. One limitation of the LinkedGeoData project is the fact that, out of a large amount of data available, only about 53,000 entities contain a direct link to a relevant DBpedia page [2, p.743]. While this matching produces good results when the entities considered are cities, it is less so for other semantic categories (e.g. countries and universities). However, in our work for map personalisation, many more categories of entities need to be taken into account.

In order to gain a better understanding of user spatial interests, this linkage between vector data and ontologies can be beneficial. Implicit feedback indicators, such as mouse movements and map navigational behaviour, can be used to infer user interests [18]. The explicit semantic associations provided by LinkedGeoData between the map presented to the user and its underlying ontological entities represent a step in this direction. Once these ontological relationships are established, it is possible to combine them with implicit indicators to enrich spatial and refine user profiles. But, in order to infer valid user interests from implicit feedback indicators such as mouse movements, a more detailed and comprehensive mapping between spatial data and ontological data is needed.

Another major topic that has not been not addressed in other projects is the impact of the map scale. When a user uses a typical interactive Web map, the semantic content of the displayed information changes depending on the scale. Our system takes this important parameter into account and allows a further exploration of the role of map scale in the user interest determination. This system, whose architecture is described in Section 3, aims at addressing these issues.

3 System Architecture

We have developed a system that contributes to fill the semantic gap in VGI data, bridging ontological concepts with geographical entities. The objective of

[4] http://dbpedia.org
[5] http://linkedgeodata.org
[6] http://www.w3.org/DesignIssues/LinkedData.html

the system is to retrieve semantic content from a geographical location, taking scale into account and finding ontological terms that can be used for user interests extraction. When the user clicks on the Web map, the system processes a spatial query mapping spatial features to semantic entities. This system has been developed as a module of our Web platform for map personalisation and visualisation outlined by McArdle et al. [20]. The module is a Web application that processes a spatial query and retrieves ontological results interacting with several Web services.

The core service of the system, which we called *Semantic Service*, is based on the Web development framework Grails[7], which provides an intuitive and effective environment for developing and deploying Web applications. Grails is also a suitable environment for creating and interacting with Web services. The company CloudMade[8] offers geographic-related Web services. Among others, a geocoding and reverse geocoding Web service that retrieve objects from the OpenStreetMap vector dataset data[9] are available. This service is particularly suitable to retrieve vector data without dealing with low level details of the underlying spatial DBMS, therefore we decided to use it as OpenStreetMap data provider.

GIS Open-Source software is one of the technologies that compose neogeography [13, p.2025]. Software packages released under licenses derived from the GNU public license have provided users and developers with increasing number of tools [24]. Similarly, one of the distinguishing aspects of VGI is the distribution of data under the Creative Commons license[10], which enables its free circulation on the Internet. As opposed to traditional 'closed' spatial datasets maintaining and licensed by a strongly vertical organisation, such open technologies offer the opportunity to increase the understanding of their properties, quality and possible usages, with an advantage for the whole community of users and developers. Therefore, we decided to adopt open technologies and data sources for this work.

Most of the open data produced by DBpedia and LinkedGeoData is stored and distributed in the Resource Description Framework (RDF). In order to manipulate those structures, our system relies on Jena, a semantic web framework for Java[11], which provides a set of functionality specific for RDF and other widely-used ontological formats. These projects offer Web services to query their datasets through the SPARQL language[12], designed specifically for RDF data, considered one of the key technologies of the Semantic Web [22]. This way complex queries can be executed remotely without maintaining a local copy of the data.

The system architecture is depicted in Figure 1. The *Semantic Service* hosts the main functionality of the system, discussed in detailed in Section 4.

---

[7] http://www.grails.org
[8] http://cloudmade.com
[9] http://developers.cloudmade.com/projects/show/geocoding-http-api
[10] http://creativecommons.org
[11] http://jena.sourceforge.net
[12] http://www.w3.org/TR/rdf-sparql-query

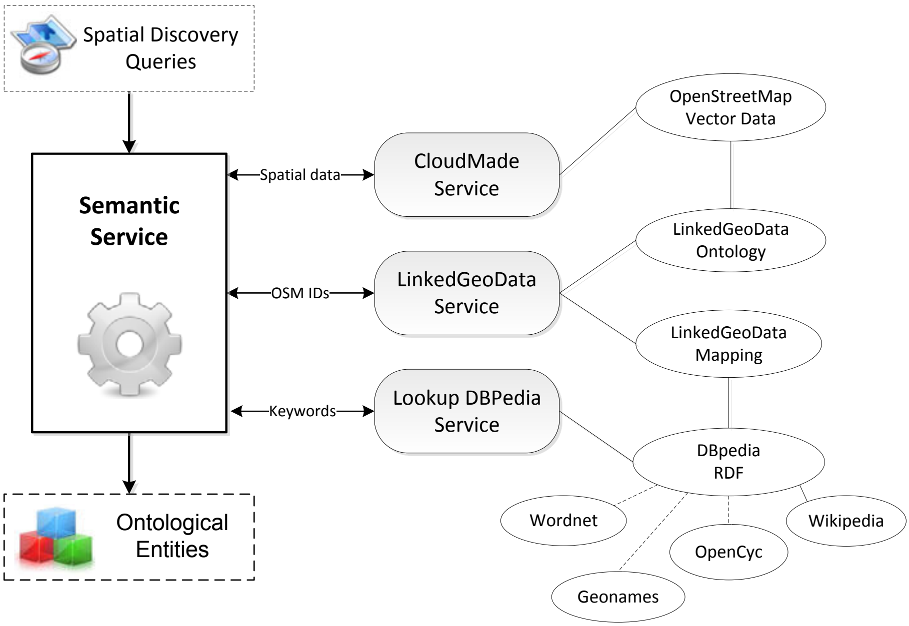


Fig. 1. System architecture

4 System Functionality

Our system aims at integrating the OpenStreetMap vector dataset (spatially rich and semantically poor) with DBpedia (spatially poor and semantically rich) in order to discover ontological concepts and entities related to a given geo-location. This is done by issuing spatial discovery queries, whose main parameters are:

- **geo-location**: latitude and longitude (e.g. *53.3071, -6.2218*);
- **radius**: radius expressed either in screen pixels (dependent on the current map scale) or in meters (e.g. *20p* or *500m*);
- **scale**: map scale (e.g. *1/14,000*);
- **max results**: maximum number of OpenStreetMap objects retrieved from CloudMade in the given area (e.g. *10*).

For example, a valid exploratory query could be:

{ *geo-location = (53.3071, -6.2218), radius = (500m), scale = (1/14,000), max results = (10)* }

This query retrieves a maximum of 10 ontological entities located within a radius of 500 metres from the specified point (expressed in the lat/long coordinates 53.3071 and -6.2218) at a low scale (1:14,000, corresponding to the street level). The indicated target geographical area roughly covers the University College

Dublin campus in Dublin, Ireland. The Semantic Service is then expected to return entities that are semantically related to the target geographical area, such as *education* as well as individual institutions located in the campus.

In this process, a critical issue is the role of the *map scale*. Commonly used Web maps (such as Google Maps and Visual Earth) take scale into account for two reasons, firstly to decide which objects need to be displayed, and secondly to choose a suitable visual style aiming to combine clarity and aesthetic coherence. In the context of implicit feedback analysis, scale plays an important role which, as Mac Aoidh et al. pointed out, has not been fully investigated [19]. Given the different visual content of a Web map at different scales, including this parameter in the semantic extraction is beneficial to further reduce the semantic gap.

A complex research question that is worth asking is: what can map scale reveal about the user's spatial interests? For example, when a user chooses a scale of 1:30,000,000, a typical Web map only renders country borders, major lakes and capital cities. If the user operates for a long time over a geographic area at a very high scale, it is reasonable to expect that they are likely to be interested into high regional features such as rivers, urban areas and major infrastructures. Considering all the entities located in the target area, such as individual cinemas and theatres, would be semantically misleading. On the contrary, when the scale is as low as 1:10,000, the user can see objects at the street level such as individual buildings, restaurants and bus stops and therefore the Web map includes all the fine-grained details available in the dataset. A project such as LinkedGeoData does not take this problem into consideration, as it only provides links for cities and few other entities.

Our system follows this idea by retrieving different objects categories at different scales, following an approach sometimes called *what you see is what you get*. The definition of these 'semantic layers' associated with different scales mirrors closely the structure of the CloudMade map, whose internal structure and style are fully customisable. We have chosen to start with the default map style, called 'the original' in the CloudMade style editor[13], because it represents the general-purpose Web maps that have become ubiquitous in neogeography. For instance, the top semantic layer in the scale range *(1:28,000,000-1:15,000,000)* only includes *countries* and *capital cities*. On the contrary the lowest semantic layer, whose scale range is *(1:2,000-1:1)*, includes categories such as *restaurants*, *ATMs* and *shops*, exposing the smallest entities contained in the dataset.

The *radius* is also an important parameter that can have a high impact on the result of the spatial query. The CloudMade service accepts the query radius in meters. In order to match the user perception more closely, our Semantic Service also accepts the radius in screen pixels, converting them into meters on the map currently being displayed. Another relevant parameter is the *maximum number* of OpenStreetMap objects to be retrieved. This parameter can be used to tune the query scope, trying to strike a balance between a more inclusive result set potentially including irrelevant entities and a smaller result set, which might exclude relevant ones.

[13] http://maps.cloudmade.com/editor

The query, including the aforementioned parameters, is processed by the Semantic Service as follows (see Figure 1):

1. *Retrieve OpenStreetMap objects from CloudMade service.*
2. *Retrieve DBpedia mapping from LinkedGeoData.*
3. *Extract key words from OpenStreetMap objects.*
4. *DBpedia lookup with extracted key words.*
5. *Heuristic to determine whether the DBpedia nodes are valid or not.*
6. *Extract ontological terms and categories.*
7. *Merge results and store them in a XML file.*

In **step 1** the system retrieves the OpenStreetMap objects located within a certain radius from the CloudMade service, taking scale into account. The OpenStreetMap nodes contain metadata and tags. For example, Figure 2 displays the OpenStreetMap node that represents University College Dublin.

```
<node id="83211617" lat="53.3071709" lon="-6.2218882"
  user="rorym" uid="23770" visible="true" version="2"
  changeset="432796" timestamp="2008-03-31T00:24:37Z">
    <tag k="amenity" v="university"/>
    <tag k="created_by" v="Potlatch 0.8a"/>
    <tag k="name:en" v="University College Dublin"/>
    <tag k="name:ga" v="An Coliste Ollscoile, Baile tha Cliath"/>
</node>
```

Fig. 2. University College Dublin in OpenStreetMap XML

It is possible to note the nature of semantic information contained in OpenStreetMap. The entity names and the *amenity* tag do not allow further semantic navigation, for example towards the concepts *education*, *school* and *college*, which are highly similar. Such data is unsuitable for implicit feedback analysis, because it does not show connections between those ontological concepts, which are useful to determine user's interests (e.g. in education). Therefore it is necessary to proceed to step 2 to get richer semantics.

In **step 2** the node ids are then extracted and matched on DBpedia nodes through the LinkedGeoData mapping dataset, described in Section 2. In the case of the University College Dublin node, the mapped entity on LinkedGeoData[14] does not contain any more information than the original OpenStreetMap node.

Afterwards, in **step 3**, key words are extracted from the node, trying to obtain useful semantic content. The extraction of key words from OpenStreetMap objects is executed by defining a subset of the tags as semantically relevant, ignoring the others. OpenStreetMap metadata, such as contributor's information and data sources (*created_by*, *user* and *source*) are discarded, as well as tags that do not seem to form points of interest for a general user (*abutters*, *smoothness*, *incline*, *voltage*). Semantically relevant tags are given a high priority in the key words list: the English name tag (*name:en*), when available, has

[14] http://linkedgeodata.org/page/node83211617

the highest priority, followed by *amenity*, *shop*, *tourism*, *landuse*, *natural*. In the case of the node representing University College Dublin, the extracted keywords are 'university', 'college' and 'dublin', which are utilised in the following step.

In order to allow users to retrieve nodes by key words, DBpedia provides a Web service called *DBpedia Lookup*[15], designed and successfully used by [16] in collaboration with the BBC. The service takes key words and returns the URI of matching DBpedia nodes, if any. In **step 4**, the Semantic Service invokes DBpedia lookup and analyses the return URIs. In the example, the service returns the University Dublin College page as a first result[16].

In **step 5** the system utilises a heuristic to determine whether the returned DBpedia node is valid or not, based on two criteria: (a) *geographic proximity* and (b) *tag matching*. To assess criterion (a), the system calculates the distance between the OpenStreetMap and the DBpedia node centroids. If the distance is lower than a threshold ϵ, the match is considered valid. A value for ϵ that seems to give reasonably good results is 50km, for example preventing frequent mismatches between European and North-American cities with the same name. In the case of University College Dublin, criterion (a) is fulfilled, with a distance smaller than 1 kilometre. When the geo-location is not available in the DBpedia node, criterion (b) is considered. All the tags present in the OpenStreetMap node are matched against the DBpedia node. If the matching tags ratio is higher than threshold σ, the node is considered to be valid. The default value of σ, based on a preliminary evaluation, is set to 0.5. The optimal values of ϵ and σ can be determined by further experimental evaluation.

In **step 6**, the retrieved valid nodes are then processed to extract ontological terms and categories, which enhance the semantic relevance of the results. For example, the DBpedia node representing University College Dublin contains, among others, the ontological term *university*[17]. By visiting this ontological term, it is possible to navigate to the parent term (*educational institution*) and, from there, to reach the terms *school* and *college*. Semantic similarities starting from the OpenStreetMap node can now be explored.

In **step 7**, all the results are merged and stored into an XML structure and can be either stored for further analysis for formatted in human-readable HTML code and displayed to a human user.

Section 5 outlines a user case which shows an interaction with the system and describes its Web GUI.

5 The Web Graphical User Interface

In this section we present the Web user interface of the system, showing an example of its functionality, and describes a preliminary evaluation. The Web GUI allows the user to execute spatial queries in a user friendly way.

[15] http://lookup.dbpedia.org
[16] http://dbpedia.org/page/University_College_Dublin
[17] http://dbpedia.org/ontology/University

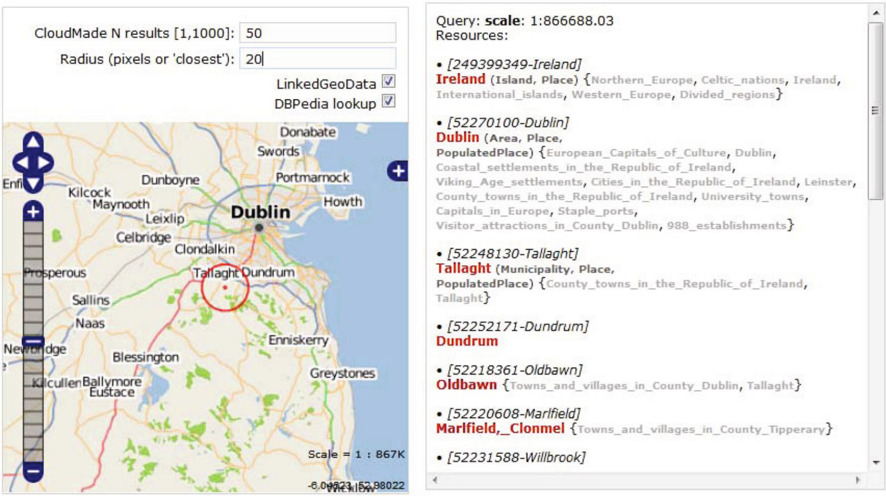


Fig. 3. Web User Interface for Spatial Queries

Discovery spatial queries can be performed on the system via an interactive Web page represented in Figure 3. This page can be used to execute spatial queries by clicking on an interactive Web map. The panel is split in two frames. The left frame contains the query parameters, which can be changed by the user, such as max number of results, radius and ontologies to be used. A CloudMade map is rendered to let the user choose a location in an intuitive way. The map (rendered with 'The Original' CloudMade style[18]). The user clicks on an area in the south of Dublin and a red circle with a red dot in its centre is displayed to represent the query area. In this query, the radius is set to 20 pixels. The map scale (about 1:860,000) is read automatically from the Web map and submitted to the server, which processes the request on-the-fly, as described in Section 4.

The query XML output is loaded in the panel in the frame on the right, converted into HTML. In this case the system has retrieved several DBpedia nodes (highlighted in bold) and has extracted ontological terms, including Ireland, Dublin and the suburbs Tallaght and Dundrum, located within the circle. Several DBpedia categories are also displayed, ranging from geographical concepts such as *Northern Europe* to political and historical ones (*Divided Regions*, *Celtic Nations*). The system has made explicit this implicit semantic content and has given to the user a list of abstract concepts contained within the selected spatial area.

Retrieving the results associated with a given location is useful to show the system capabilities, but it cannot give any insight on the properties of the datasets. In order to analyse the system behaviour on large geographical areas, we have built an interface to perform sampling experiments.

[18] http://maps.cloudmade.com/editor

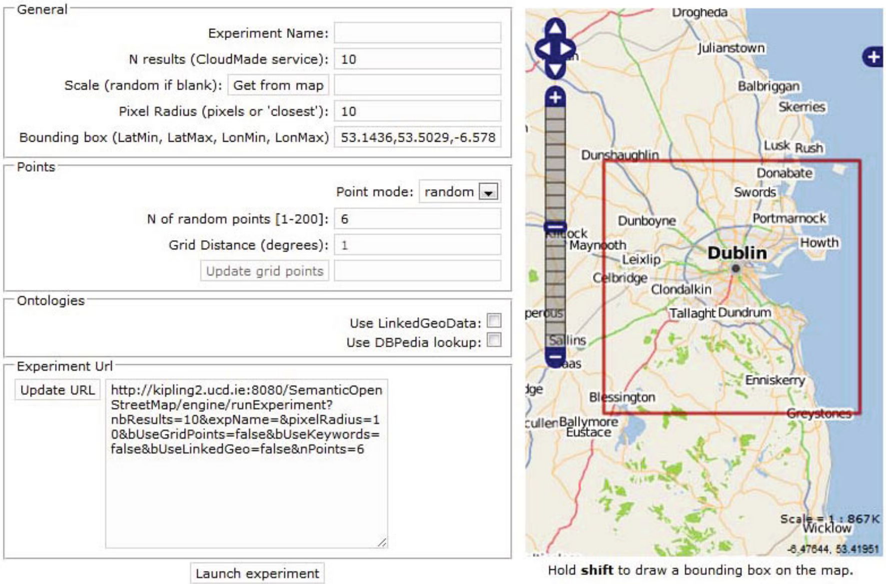


Fig. 4. GUI for sampling a bounding box by running multiple queries

Through the page represented in Figure 4 the system can sample the datasets. The user can draw a bounding box on the map and choose several parameters of the sampling to be performed. The controls in the *ontologies* section of the page allows the user to select which ontologies have to be included in the experiment. In order to contrast the results, it is possible to enable or disable the LinkedGeoData mapping and the DBpedia lookup service.

The points to be analysed can be selected in 2 modes: *random* and *grid*. The former consists of selecting a number of random locations within the bounding box, while the latter distributes points on the bounding box at a regular distance (specified in the parameter *grid distance*, expressed in degrees). The map scale can either be set to a given value for all the points, or can be selected randomly for each point. This way it is also possible to observe what the impact of the scale is on the results.

The experiment can be executed with the button *launch experiment*. The system then executes a query on each point with the selected parameters, and all the results are merged. For the moment the complexity of the procedure is linear, so the time grows linearly with the amount of selected points. When completed, the experiment results are stored in the XML file containing all the spatial queries and the relevant ontological entities. The input and output of each query are included in the file, in order to be easily machine-processed and analysed. At the end of the procedure, the URI of the XML document is returned to the user, and the file can be used for any other analysis.

Figure 5 shows an example of *grid* sampling of a large bounding on the Irish East coast with 12 points. When executing a small experiment with scale set

Fig. 5. Grid sampling a bounding box surrounding the Dublin area

```
<resource index='0'>
  <osm id='52270446' name='Roundwood'>
          <location lon='-6.22481' lat='53.06347' />
      </osm>
      <dbpedia>
          <node>Roundwood</node>
          <location lon='-6.2333' lat='53.06' />
          <subjects>
            <category>Towns_and_villages_in_County_Wicklow</category>
          </subjects>
          <ontology>
             <term>Municipality</term>
             <term>Place</term>
             <term>PopulatedPlace</term>
          </ontology>
  </dbpedia>
</resource>
```

Fig. 6. A sample of the XML code returned by the system

to 14,000, radius equal to 10 pixels (equivalent to 50 meters at this scale), the results contain 13 DBpedia nodes, 43 categories and 27 ontological terms. One of the retrieved resources is the Irish village Roundwood, stored in the XML code in Figure 6 contains an OpenStreetMap structure (*osm*) and a DBpedia node (*dbpedia*), which contains a geo-location, one category and 3 ontological terms. This code can be easily processed and the relevant URI[19] can be accessed. Thus, links between the spatial vector data displayed to the user and the ontological and semantic richness of DBpedia have been enabled.

During the development of the system we carried out a preliminary evaluation with postgraduate students of the School of Computer Science and Informatics (University College Dublin). In order to engineer the heuristics, establish a suitable value for the thresholds and identify issues, we defined three alternative versions of the system, combining the phases in different ways:

[19] http://dbpedia.org/resource/Roundwood

- *Version 1*: OpenStreetMap + LinkedGeoData mapping
- *Version 2*: OpenStreetMap + Our heuristics
- *Version 3*: OpenStreetMap + LinkedGeoData mapping + Our heuristics

Version 1 and 2 are using only certain parts of the system, while Version 3 exploits its full functionality, similarly to the approach used by Mirizzi et al. [21]. The following parameters have been defined as constant: ϵ (maximum node distance) = 50km, σ (tag matching ratio) = 0.5, *maximum results* (CloudMade service) = 10, *radius* = 20 pixels, *scale* = random, *bounding box* = Dublin area.

The users have been presented with a set of 6 random spatial queries and the correspondent view on the CloudMade map scaled and centred to the query location. The queries had been performed with different versions of the algorithm (2 for each version), hiding this information from the user. The users have then been asked to rank the semantic relevance of each node (either DBpedia entity or ontological term) on a Likert scale from 1 to 5 (from *strongly uncorrelated* to *strongly correlated* with the visible map).

During this preliminary evaluation, several issues have been identified. Very little difference separates the performance of Version 2 and 3. The LinkedGeoData mapping, although very accurate when present, did not significantly improve the results, which were mostly return by our heuristics based on DBpedia lookup. In some cases, such as for the term *Smithfield*[20], which identifies numerous places in former British colonies, the heuristics succeeded and selected the correct neighborhood in Dublin, based on the geo-location. However, certain mismatches still occurred. When the OpenStreetMap feature contains a very frequent term in a certain geographical area and does not have other semantic content, a mismatch is likely to occur. For example, this typically happens with common surnames (e.g. *Smith*) that are highly frequent in the datasets. Overall, on 47 retrieved nodes, 4 were considered as irrelevant (8.5%).

Although this preliminary evaluation has confirmed that the system works correctly in most cases, the sample is too small to draw general conclusions about it. The objective of this evaluation was mainly to identify design flaws that need to be addressed. In order to draw general conclusions about the system performance, an extensive evaluation is needed. In particular, more independent variables need to be defined, such as the thresholds, the radius and the maximum number of results, and a bigger sample of queries must be taken into account. This way, several aspects of the system behaviour can be better assessed and quantified.

Section 6 draws conclusions and outlines our plans for system evaluation and future work.

6 Conclusions and Future Work

In this paper we have described a system that aims to fill the semantic gap between spatial and non-spatial user-generated data. Starting from a semantically

[20] http://dbpedia.org/page/Smithfield

poor dataset (OpenStreetMap), the system performs spatial discovery queries and expands the semantic content of a given geographic location. This process relies on the LinkedGeoData dataset mapping and and extends it with the addition of key word extraction from OpenStreetMap nodes. Furthermore, through a novel heuristic technique, the system filters out uncorrelated entities. The system retrieves matching DBpedia nodes (semantically rich, spatially poor), collecting ontological knowledge related to the given location. This way semantic relationships between a geographical location and ontological concepts are established and can refine implicit feedback analysis and enable other applications. In this process the system takes into account the map scale at which the user is working for improved relevance during retrieval.

Although our preliminary evaluation seems promising, an extensive evaluation needs to be carried out in order to gain further understanding of the system strengths and weaknesses. The technique implemented in the system relies on several thresholds, described in Section 4. The optimal values of these thresholds should be established by experimental evaluation, based on performance metrics. A fundamental metric in this context is the relevance of related entities to human users, which was used in our preliminary evaluation described in Section 5. Defining query parameters (scale, geographical region, radius, etc) as independent variables, it is possible find out which conditions impact on the system performance and why.

Another area that needs further exploration is the navigation of ontological entities and concepts, starting from the ones returned by the system, which do not necessarily include all relevant concepts. A body of work exists on similarity metrics between DBpedia nodes. Kobilarov et al. utilises a metric based on distance in the DBpedia graph [16], while Hassanzadeh and Consens investigate several similarity measures based on an approximate string matching [15]. Starting from a set of nodes returned by the system, it is possible to explore similar concepts, showing correlation between geographically and semantically close entities and see how this relates to the user interests.

From a more quantitative viewpoint, it would be interesting to use the *experiment* interface to study and contrast the datasets spatial and semantic statistical properties. Comparing the system results in different geographical areas (e.g. Europe vs North-America) and semantic areas (e.g. human-built entities vs natural entities) can give insight not only on the system performances, but also on the properties of these open datasets. We plan to further investigate the possibilities of implicit feedback analysis and spatial personalisation offered by such open spatial datasets and ontologies, whose visibility and impact are bound to increase in the near future, in academia and industry alike.

Acknowledgements

Research presented in this paper was funded by a Strategic Research Cluster grant (07/SRC/I1168) by Science Foundation Ireland under the National Development Plan. The authors gratefully acknowledge this support and thank Paul Huynh for his contribution to the system implementation.

References

1. Auer, S., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R., Ives, Z.: DBpedia: A nucleus for a web of open data. In: Aberer, K., Choi, K.-S., Noy, N., Allemang, D., Lee, K.-I., Nixon, L.J.B., Golbeck, J., Mika, P., Maynard, D., Mizoguchi, R., Schreiber, G., Cudré-Mauroux, P. (eds.) ASWC 2007 and ISWC 2007. LNCS, vol. 4825, pp. 722–735. Springer, Heidelberg (2007)
2. Auer, S., Lehmann, J., Hellmann, S.: LinkedGeoData: Adding a spatial dimension to the web of data. In: Bernstein, A., Karger, D.R., Heath, T., Feigenbaum, L., Maynard, D., Motta, E., Thirunarayan, K. (eds.) ISWC 2009. LNCS, vol. 5823, pp. 731–746. Springer, Heidelberg (2009)
3. Berners-Lee, T., Hendler, J., Lassila, O., et al.: The semantic web. Scientific American 284(5), 28–37 (2001)
4. Bishr, M., Kuhn, W.: Geospatial Information Bottom-Up: A Matter of Trust and Semantics. In: AGILE 2007. pp. 365–387. Springer, Denmark (2007)
5. Bizer, C., Heath, T., Berners-Lee, T.: Linked Data – The Story So Far. International Journal on Semantic Web and Information Systems 5(3), 1–22 (2009)
6. Bizer, C., Lehmann, J., Kobilarov, G., Auer, S., Becker, C., Cyganiak, R., Hellmann, S.: DBpedia – A crystallization point for the Web of Data. Journal of Web Semantics 7(3), 154–165 (2009)
7. Ding, L., Kolari, P., Ding, Z., Avancha, S.: Using ontologies in the Semantic Web: A survey. ontologies, 79–113 (2007)
8. Flanagin, A., Metzger, M.: The credibility of volunteered geographic information. GeoJournal 72(3), 137–148 (2008)
9. Fonseca, F., Egenhofer, M.: Ontology-Driven Geographic Information Systems. In: Proceedings of the 7th ACM International Symposium on Advances in Geographic Information Systems, pp. 14–19. ACM-GIS (1999)
10. Fonseca, F., Egenhofer, M., Agouris, P., Câmara, G.: Using ontologies for integrated geographic information systems. Transactions in GIS 6(3), 231–257 (2002)
11. Goodchild, M.: Citizens as sensors: The world of volunteered geography. GeoJournal 69(4), 211–221 (2007)
12. Haav, H., Kaljuvee, A., Luts, M., Vajakas, T.: Ontology-Based Retrieval of Spatially Related Objects for Location Based Services. In: Meersman, R., Dillon, T., Herrero, P. (eds.) OTM 2009. LNCS, vol. 5871, pp. 1010–1024. Springer, Heidelberg (2009)
13. Haklay, M., Singleton, A., Parker, C.: Web Mapping 2.0: The Neogeography of the GeoWeb. Geography Compass 2(6), 2011–2039 (2008)
14. Haklay, M., Weber, P.: OpenStreetMap: User-Generated Street Maps. IEEE Pervasive Computing 7(4), 12–18 (2008)
15. Hassanzadeh, O., Consens, M.: Linked Movie Data Base. In: Proceedings of the 2nd Workshop on Linked Data on the Web (LDOW 2009), Madrid, Spain (2009)
16. Kobilarov, G., Scott, T., Raimond, Y., Oliver, S., Sizemore, C., Smethurst, M., Bizer, C., Lee, R.: Media meets semantic web – how the BBC uses dBpedia and linked data to make connections. In: Aroyo, L., Traverso, P., Ciravegna, F., Cimiano, P., Heath, T., Hyvönen, E., Mizoguchi, R., Oren, E., Sabou, M., Simperl, E. (eds.) ESWC 2009. LNCS, vol. 5554, pp. 723–737. Springer, Heidelberg (2009)
17. Korpipää, P., Häkkilä, J., Kela, J., Ronkainen, S., Känsälä, I.: Utilising context ontology in mobile device application personalisation. In: Proceedings of the 3rd International Conference on Mobile and Ubiquitous Multimedia, pp. 133–140. ACM, New York (2004)

18. Mac Aoidh, E., Bertolotto, M., Wilson, D.: Analysis of implicit interest indicators for spatial data. In: Proceedings of the 15th Annual ACM International Symposium on Advances in Geographic Information Systems, p. 47. ACM, New York (2007)
19. Mac Aoidh, E., Bertolotto, M., Wilson, D.: Understanding geospatial interests by visualizing map interaction behavior. Information Visualization 7(3), 275–286 (2008)
20. McArdle, G., Ballatore, A., Tahir, A., Bertolotto, M.: An Open-Source Web Architecture for Adaptive Location Based Services. In: Proceedings of the 14th International Symposium on Spatial Data Handling (SDH), Hong Kong, vol. 38(2), pp. 296–301 (2010)
21. Mirizzi, R., Ragone, A., Di Noia, T., Di Sciascio, E.: Ranking the linked data: The case of dBpedia. In: Benatallah, B., Casati, F., Kappel, G., Rossi, G. (eds.) ICWE 2010. LNCS, vol. 6189, pp. 337–354. Springer, Heidelberg (2010)
22. Prud'hommeaux, E., Seaborne, A.: SPARQL query language for RDF (2006), http://www.w3.org/TR/rdf-sparql-query
23. Sieg, A., Mobasher, B., Burke, R.: Web Search Personalization with Ontological User Profiles. In: Proceedings of the 16th ACM International Conference on Information and Knowledge, pp. 525–534. ACM, New York (2007)
24. Steiniger, S., Bocher, E.: An Overview on Current Free and Open Source Desktop GIS Developments. International Journal of Geographical Information Science 23(10), 1345–1370 (2009)
25. Turner, A.: Introduction to Neogeography. O'Reilly Media, Inc., Sebastopol (2006)
26. Völkel, M., Krötzsch, M., Vrandecic, D., Haller, H., Studer, R.: Semantic Wikipedia. In: Proceedings of the 15th International Conference on World Wide Web, Edinburgh, Scotland, pp. 585–594. ACM, New York (2006)
27. Yu, S., Spaccapietra, S., Cullot, N., Aufaure, M.: User Profiles in Location-Based Services: Make Humans More Nomadic and Personalized. In: Proceedings of the IASTED International Conference on Databases and Applications, Innsbruck, Austria. ACTA Press (2004)

A Mobile Navigation and Orientation System for Blind Users in a Metrobus Environment

Felix Mata[1], Andres Jaramillo[2], and Christophe Claramunt[3]

[1,2] National Polytechnic Insititute –UPIITA
Mobile Computing Laboratory, Av. IPN 2580 , Mexico, DF
mmatar@ipn.mx
[3] Naval Academy Research Institute
Lanveoc-Poulmic, 29240 Brest Naval, France
claramunt@ecole-navale.fr

Abstract. The research presented in this paper introduces a mobile assistant to spatially locate and orient passengers of a Metrobus system in the city of Mexico. The system assists blind passengers or passengers with limited eyesight. While most of the navigation systems developed so far for blind people employ a complex conjunction of positioning systems, video cameras, location-based and image processing algorithms, we developed an affordable and low-cost combination of mobile phone, GPS and compass device to provide the same range of functionalities. The mobile application is developed on top of a 3rd generation cell phone extended by GPS and digital compass devices. Interaction among these devices is achieved by using Bluetooth communications. Audible user-oriented interfaces indicate to a given user her/his location within the Metrobus system, i.e., appropriate instructions of orientation that lead to the boarding gates in that station, thanks to an orientation algorithm developed for this aim. The experiments have been performed on a Nokia N73 cell phone in a Metrobus line of the city of Mexico and validated by a panel of blind users.

Keywords: Orientation Algorithm, mobile navigation, blind users.

1 Introduction

Despite continuous progress, navigation systems and services still need further developments in order to fulfill human spatial and cognitive abilities. Modeling navigation knowledge and tasks require an integration of several large-scale concepts such as landmarks and route directions in order to provide a better sense of the environment to the users. This is closely related to the ability of humans to acquire a holistic and functional view of large-scale spaces. For blind people, it has been recognized that they have the capacity of memorizing a given environment, but they need more time to acquire it [11]. Generally, blind people rely on intensive exploration of a given environment to derive spatial knowledge [9], or more commonly tier assistance.

Over the past few years several technological developments have provided novel opportunities to assist blind people in the interpretation of cartographical information and navigation tasks [10]. For instance, audio-based services have been recently

K. Tanaka, P. Fröhlich, and K.-S. Kim (Eds.): W2GIS 2011, LNCS 6574, pp. 94–108, 2011.

explored to provide interfaces to mapping systems [15]. The objective of these systems is to mediate a sense of the environment using speech messages to blind people. The idea behind is to provide to the users an holistic view of the environment, by relating their locations to objects of interest in their neighborhood, then helping them to act and move. Also, cell phones available for blind people have been extended by several interface innovations. For instance, blind people use cellular phones with voice commands, audible menus and interfaces, whose objective is to replace graphic or textual messages by speech messages. Those have been integrated by many mobile phone software that offer text to speech functions such as the names of incoming calls, number of missed calls, and other statistics and services available in the telecommunication network [3]. One of the specific characteristics of cell phones relies in the fact that most of the users are mobile, thus making a closer integration with GPS systems a valuable extension to their functionalities. What one should ideally expect from such systems are novel opportunities to help blind people to locate and orientate themselves when navigating. It has been shown that qualitative direction relations are particularly important when a user is interacting and acting in a given environment, as well as qualitative distances to a less degree [4]. When combined, 3rd or 4th generation cell phones, GPS and digital compass devices might provide the technological setup for the development of mobile assistant systems for blind people navigating in the environment. For instance, when locating a place with a GIS an the direction to that place with a digital compass, a cell phone can guide step by step a user to find and move to that place.

The research presented in this paper introduces a mobile assistant for orienting blind people in a *Metrobus* environment. The computing framework is developed on top of a low-cost 3rd generation cell phone extended by GPS and digital compass devices. Bluetooth communications support interactions between the different devices. The interfaces developed are audio-based and convert spatial information to appropriate speech instructions and messages. The principle of the mobile assistant is to first identify the location of a blind user in the *Metrobus* system in order to give her/him appropriate direction instructions according to her/his navigation tasks (e.g., station exit, boarding gates).

The experimental setup developed has been applied to a transportation facility of the line number one of the *Metrobus* of the city of Mexico. Mexico has around 2.2 million people with physical disabilities, amongst them, 28.6% are blind or only see shadows, 16.5% are deaf or hear using apparatus [8]. The line number one of the *Metrobus* has an average of 260 thousand passengers on a working day. Stations and buses are not entirely suitable for a blind person to move safely. Only a few stations have plates with information in Braille and others guides. As others, blind people use the *Metrobus* although this activity is far from been straightforward, and sometimes even a dangerous activity. For example, when blind people use the *Metrobus* system, it is difficult for them to act in a given station, such as to identify the direction to get the correct boarding gate or to find the exit. Therefore, they are often delayed and require other people to help them, which further complicate their behavior.

The remainder of the paper is organized as follows. Section 2 introduces related work in the development of mobile assistant for blind people. The principles of our approach are developed in Section 3, while experimental results are presented in Section 4. Finally Section 5 concludes the papers and outlines some research perspectives.

2 Related Work

Most of the developments so far achieved for helping blind people in navigation tasks combine research progress in cognitive and psychological sciences with technological innovations. Experiments have been either directed to indoor or outdoor environments, and taking into account the fact that blind people have different knowledge of the environment when they act in either a well-known, partially known or unknown surroundings.

An assistant system for blind users has been developed where the goal is to enhance interactions between a blind person and technological devices at home [16]. The objective is to encourage blind persons to interact with every day technological artifacts and control the different functions of their home. The objective is to favor user independence and even socialization. When interacting in indoor environments, a sensor-based system that assists blind users in orienting tasks has been introduced [6]. This sensor is used as the equivalent of a directed light for searching targets. A blind user presses appropriate speech-based keys for inquiries concerning object characteristics, position and orientation.

Elsewhere, a tactile display provides a non-visual support for way finding that guides and keeps a mobile user in route [5]. A vibrator belt indicates directions and deviations from the path in an accurate and unobtrusive way. A guidance system for people with visual disabilities have been developed and implemented in the city of Neuhausen in Switzerland. RFID communication identifies the different stops of a given bus line, and react to the radio signals transmitted by the buses in order to transmit stop requests and open doors when required [18]. However, navigation instructions are not provided. Similar applications have been developed on top of PDAs, where RFID tags are used to locate people, but the infrastructure required is expensive and not completely implemented [13, 19]. More recently, several guidance systems have been developed on top of smart phones and GPS, but still these systems are at the trial stage [14, 17]. A guidance travel system that combines mobile phones with a camera has been experimented in the airport of Zurich. The objective is to provide route directions, but still it is based on complex location algorithms, specific infrastructure in the indoor environment, and image processing for the detection of landmarks in the environment [1], this making the whole process a non straightforward task [7].

The mobile assistant developed in this paper differs in many aspects from the ones presented above. First it combines location and direction instructions, secondly it is developed on top of a low-cost mobile computing and telecommunication infrastructure that favors its implementation in a large metropolitan city such as Mexico.

3 Methodology

The main objective of our mobile assistant is to orient and locate people with impaired vision in several stations of the *Metrobus* transport system. The mobile assistant is based on audible menus whose objective is to give locations (e.g., “you are in the Insurgentes station”) and orientation instructions (e.g., “the exit is at the right”). The system runs on a cellular phone in conjunction with a GPS and a digital compass.

The *Metrobus* line 1 has thirty-three stations (e.g. Indios Verdes, Insurgentes, Dr. Gálvez) and three terminals (North terminal, Center terminal, South terminal). Station platforms have a height of one meter above the street level. The infrastructure was designed in order to allow access to people with special abilities [7]. Figure 1 shows an example of a typical *Metrobus* station.

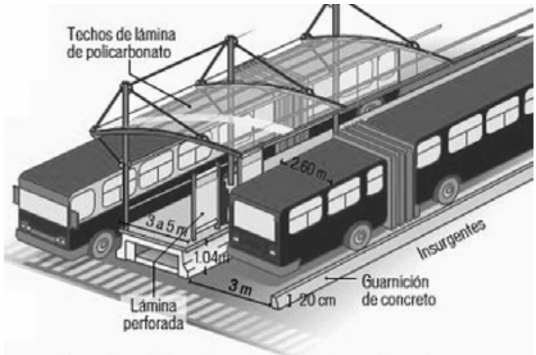


Fig. 1. A Metrobus station

As shown in Figure 1, the *Metrobus* runs on an elevated structure without major interferences and can receive GPS signals without problems. The instructions given to a blind user should give her/him the direction to either the boarding gates or the exit. Location (e.g., you are located in Dr. Galvez station) and orientation messages (e.g., the boarding gates are located at the right, the exist is located straight ahead) are supported by an audible interface.

Fig. 2. Framework principles

Figure 2 introduces the main principles of the mobile assistant framework. The mobile assistant is developed on top of a cell phone with compass and GPS included. The mobile assistant offers the possibility to orient and locate exits, giving direction and distance information to blind users using audible menus. The overall functionality is provided when a user is located within or nearby one of the *Metrobus* stations. The functionalities of the mobile assistant are organized into three main components:

1) an orientation and location scheme,

2) an orientation algorithm,

3) audible menus.

The functionalities of these components are developed in subsections 3.1, 3.2 and 3.3, respectively.

3.1 Orientation and Location Schemes

When starting the initialization phase, the mobile assistant requires location and directional information previously stored on the mobile phone. We developed an orientation and location scheme, where the GPS coordinates and cardinal directions of the main elements of each *Metrobus* station are captured and stored into a reference XML database which is locally stored in the mobile assistant.

We take as relative reference the centre of each *Metrobus* station to capture cardinal directions using a GPS and a magnetic compass device. Figures 3 and 4 illustrate the process.

Fig. 3. Geographic coordinates in a Metrobus station

Location information are labeled and stored into a XML database whose structure is illustrated in Table 1.

Table 1. Station identifiers and geographic coordinates

Record ID	*Metrobus* Station	Latitude	Longitude	Station Identifier
16	Tabacalera	19,43609	99,15737	161185935
17	Reforma	19,43299	99,15867	171185917
18	Hamburgo	19,42780	99,1611	181185889

Let us introduce the main principles of the directional referencing process. Suppose that a blind user located in a given *Metrobus* station wants to know where is the exit located. This implies to know

- the location of the user in that *Metrobus* station (step 1),
- towards which cardinal point is oriented the exit(s) of the station (step 2),
- towards what cardinal point is pointing at the user (step 3).

Step 1 is given by the GPS (i.e. the location of the user and implicitly her/his relative location with respect to the center of the station). Step 2 requires to capture and record the cardinal directions of the exit with respect to the center of all *Metrobus* stations (Figure 4).

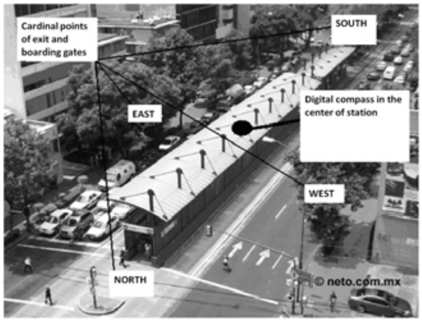


Fig. 4. Cardinal directions in a *Metrobus* station

Cardinal directions of boarding gates and exits are collected relative to the geographical center of each station. Table 2 shows a sample of the obtained lectures by digital compass in some of the stations and typical values of cardinal directions for boarding gates and exits (North, South, etc) for several *Metrobus* stations.

Table 2. Cardinal values for *Metrobus* stations

Number	Station	Boarding gates direction: *Indios Verdes*	Boarding gates direction: Dr. Gálvez	Exit
1	Indios Verdes	-----------	E: ***04***	S: ***08***
2	18 de Marzo	-----------	NW,W ***14,12***	N ***16***
3	18 de Marzo	SE,E ***04,06***	-----------	N ***16***
4	Euskaro	-----------	NW,W ***14,12***	S ***08***
10	Circuito	SE,E ***04,06***	-----------	NE ***02***
11	San Simón	SE,E ***04,06***	NW,W ***14,12***	S ***08***
12	Manuel González	SE,E ***04,06***	NW,W ***14,12***	NE ***02***

3.2 Orientation Algorithm

The functionalities of the mobile assistant are based on an orientation and location algorithm that defines which audible instruction will be provided to the user (Steps 2 and 3). The algorithm is based on orientation directions converted into clockwise directions. This is based on the observation that blind people commonly use a clockwise notation to orient themselves [12]. For example, when a blind people want to say that something is located "at the right", she/he might use the following expression "at three o'clock". Similarly, "at six o'clock" means "behind". Blind persons often use these expressions to locate objects and places in the environment. Clockwise directions were recorded in order to be played appropriately by the mobile assistant together with target instructions. The calculation process that gives orientation instructions consists of two parts (Step 3). The objective of the first one is to obtain the user location and calculate her/his distance to the reference point (i.e., center of station) in order to infer if the searched targets are located at the same direction that the one of the center of the station, or more likely if that direction is different when the user is not located in the center of the station.

Table 3. Clockwise and audio expressions

Cardinal direction	Directional expression	Audio expression recorded
(N) North	In front	At twelve o'clock
NNW	-	At eleven o'clock
(NW) Northwest	Between left and in front	At half past ten
NWW	-	At ten o'clock
West	At left	At nine o'clock
SWW	-	At eight o'clock
(SW) South West	Between left and be-hind	At half past seven
SSW	-	At seven o'clock
(S) South	Behind	At six o'clock
SSE	-	At five o'clock
(SE) Southeast	Between right and be-hind	At half past four
SEE	-	At four o'clock
(E) East	At right	At three o'clock
NEE	-	At two o'clock
(NE) Northeast	Between right and in front	At half past one
NNE	-	At one o'clock

The second part of Step 3 considers the directions to which the user is pointing at with her/his cell phone. Overall, the respective roles of these direction relations, locations of the user and center of the station, are illustrated in Figure 5, and where

- α denotes the angle between the main direction of the station and the target of interest of the *Metrobus* station (e.g., boarding gates, exit),
- γ denotes the angle between the the North and the main direction of the *Metrobus* station,
- β denotes the angle between the direction of the user pointing and the North,
- ∂ denotes the angle between the target of interest of the Metrobus station and the user pointing.

The angle α between the user and the target of the Metrobus station is approximated as follows

$$\alpha = ArcTan(d1/d2) \tag{1}$$

and the angle ∂ between the user pointing and the target of the *Metrobus* station is given by

$$\partial = (\alpha + \gamma + \beta) \tag{2}$$

where the final clockwise directions to the target of interest given to the user by the mobile assistant are illustrated in Table 3.

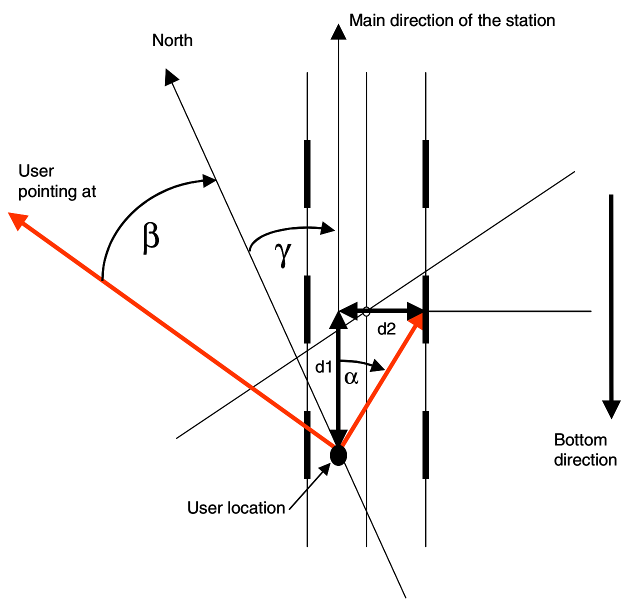


Fig. 5. Orientation scenery within a *Metrobus* station

All static data are locally stored in the mobile assistant as an XML database. The only information which is user-dependent and calculated on request is the location of the user and the direction her/his pointing at. We illustrate the application of the orientation algorithm by considering four different users located in a Metrobus station. These users are pointing to different cardinal directions: North, South, West, and East, respectively (Figure 6).

Let us first consider the case depicted by Figure 6 and Table 3, where *user1* is located in San Simon station and wants to know at what directions are located the boarding gates towards *Indios Verdes* terminal (West, "nine o 'clock"). Let us also consider that *user1* is pointing to the South (i.e., at six o'clock).

First, the center of the station is known and given by the coordinates of a reference point denoted as point1 (19.45968, 99.14638). From the center of the station, the two boarding gates towards *Indios Verdes* terminal are located at the North West and at the West (see Table 2), respectively (initial values from the center of all stations are stored into the cell phone).

Secondly, the location of the user, denoted as point2 (19.45973, 99.14640), is given by the GPS. Next, the distance between these two points is evaluated. In this case, that user is located at a distance d2=5.9 meters from the center of station, where the distance of the center of station to the central boarding gate to the bottom direction is d1=2.5 meters. By application of formula 1 and 2, the correct direction to give to the user should be "Boarding gates towards *Indios Verdes* terminal are at three o'clock" (that is "at right").

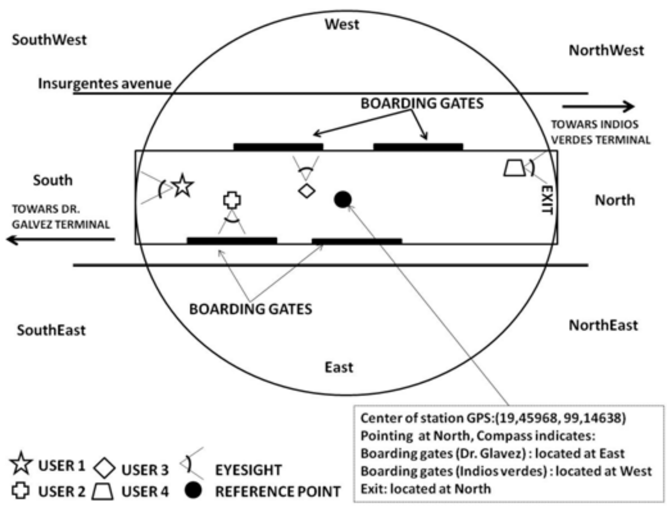


Fig. 6. Testing scenery in a *Metrobus* station

Let us now consider that user1 is pointing to a cardinal direction β = South, at six o'clock. The cardinal direction of the boarding gates is as follows: boarding gate1 (NWW) and boarding gate2 (NNW). Then, application of formula 1 and 2 gives the clockwise directions of the boarding gates, which are converted and played as audio expression as follows

> "Boarding gates towards *Indios Verdes* terminal are located at four o'clock and five o'clock"

This process is similarly applied to the other cases according to the scenery introduced in Figure 7. For user2 we have the following configuration:

- center of the station (19.45968, 99.14638), user2 location (19.45966, 99.14639), the target searched is the exit located at the North and user2 is pointing to the East.
- The answer given by formulae 1 and 2 is "The exit is at nine o'clock"

For user3 we have the following configuration:

- center of station (19.45968, 99.14638), user3 location (19.45967, 99.14641), the target searched is the exit, at the North and user3 is pointing to the East, at three o'clock.
- The answer given by formulae 1 and 2 is "The exit is at three o'clock".

Finally, when considering user4 we have

- center of station (19.45968, 99.14638), user4 location (19.45977, 99.14641), the target searched is the exit at the South East, user4 is pointing to the North.
- The answer given is a concatenation of audio files in order to play: "Boarding gates towards *Dr. Galvez* terminal are located at half past seven o' clock and six o'clock.

3.3 Audible Menus

The third and last step of the methodology involves the design of the audible interfaces. These interfaces have been designed using audio files that play directional-location instructions to the blind users. The functional interaction is as follows: a user browses the mobile assistant application by pressing buttons keys and listening audio files according to the options available (blind users are familiarized with this sort of environment and the use of menu audios). The main principles of these audible interfaces are as follows: each item of a menu plays an audio file (i.e., name of the menu) to indicate which or at what level of a menu the user is interacting with. The user can browse through the different options of the menus using appropriate keys. The design of the audible interfaces was developed to answer the following examples and categories of queries:

- Query 1: At what station is the user located?
- Query 2: At what directions are the exit or exits located?
- Query 3: At what directions are the boarding gates for a given terminal located?

Each query is answered by listening a set of audios and browsing menus and options. For example, query 1 is answered by selecting the station of the main menu, and the option "Name" from the station menu. Next, the location of the user is given by the GPS, and the direction value obtained from the compass is processed using formula 1 and 2, then the corresponding audio file is returned, and last the audio file that says the name of the station is played. The user can repeat the audio files or go back to the previous menu by listening the available options. Another example is given by query 2 (cf. Figure 6) where the exit of station is located towards the North. The system finds that the exit of station is at the right, and this result is played in audio format (the audio says: “The exit is located at three o’clock”). The process is similar to the rest of the options and menus. Interaction between the orientation algorithms is maintained using the architecture request-response. In that form, the last phase of the mobile assistant is terminated.

4 Experiments and Results

The whole approach has been experimented using a Nokia N73 cell phone, GPS G – Trendand and a CMPS03 compass device [2]. Figure 7 illustrates the experimental setup of a user navigating within a *Metrobus* station assisted by the mobile assistant, and where that user has in hands the GPS, compass, cell phone, and audio-phones.

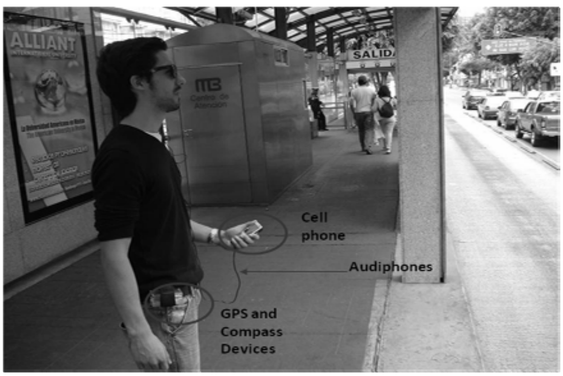


Fig. 7. A User using the GPS and compass device within a Metrobus station

Twenty persons divided into two groups have tested the system in the Metrobus stations. People with impaired vision compose the first group while blind people form the second group. In particular, the first group is formed by ten persons with impaired vision, and which commonly browse by audible interfaces and are assisted by marks provided in key number nine in the majority of cell phones. Moreover, four persons of the first group were familiar with the environment of the Metrobus station, the others are unfamiliar. Regarding the second group, three persons were familiar with the environment. The range of ages for both groups was eighteen to fifty, and all of them had a previous experience with the use of phone assistants. Last, the users practiced the mobile assistant during a displacement in the *Metrobus* line 1.

Figure 8 illustrates several examples of typical user trajectories with and without assistance of the mobile assistant. Those trajectories are depicted for two cases similar to the ones depicted by user1 and user4 according to the previous configurations presented in the orientation tasks. These examples represent the sort of behavior generated by the mobile assistant.

All users were requested to test the mobile assistant two or three times before experimenting it for the evaluation. For evaluation and testing purposes, the area of the

Metrobus station is divided into a grid of eight sections to evaluate the precision of the orientation instructions and trajectories followed by the users (Figure 8). According to the configuration, areas labeled with numbers represent exits and entrances of stations, while the ones labeled with letters denote boarding gates.

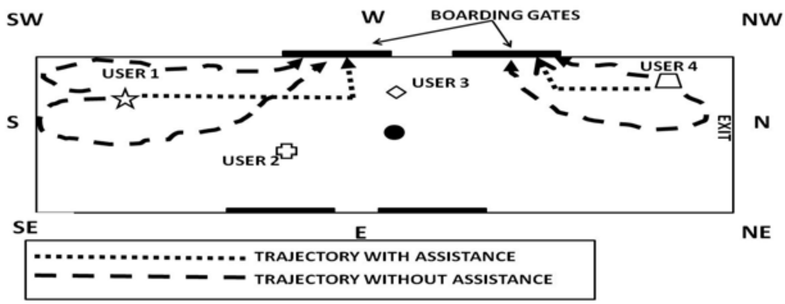


Fig. 8. Trajectories and zones within a *Metrobus* station

Table 3 averages the differences in angle for orientation instructions obtained by people with impaired vision.

Table 3. Results for group 1: people with impaired vision

	From a point in zone 1 (difference of direction)		From a point in zone 2 (difference of direction)		From a point in zone 3 (difference of direction)		From a point in zone 4 (difference of direction)	
	With assistant	Without assistant	With assistant	Without assistant	With assistant	Without assistant	With assistant	Without assistant
To point in zone A	3^0	30^0	4^0	28^0	4^0	27^0	5^0	28^0
To point in zone B	5^0	26^0	6^0	23^0	3^0	24^0	4^0	22^0
To point in zone C	6^0	25^0	7^0	26^0	5^0	23^0	5^0	26^0
To point in zone D	3^0	22^0	4^0	23^0	2^0	22^0	1^0	24^0

Table 4 shows the results for the second group made of ten blind persons.

Table 4. Results for group 2: blind people

	From point in zone 1		From point in zone 2		From point in zone 3		From point in zone 4	
	With assistant	Without assistant	With assistant	Without assistant	With assistant	Without assistant	With assistant	Without assistant
To point in zone A	1^0	60^0	2^0	75^0	3^0	64^0	2^0	66^0
To point in zone B	3^0	59^0	3^0	58^0	2^0	70^0	2^0	87^0
To point in zone C	4^0	71^0	4^0	60^0	2^0	55^0	3^0	92^0
To point in zone D	2^0	120^0	2^0	72^0	1^0	70^0	1^0	98^0

As clearly shown by Figure 8 and the differences in angle instructions for the two groups, navigation instructions provided by the mobile assistant are overall well interpreted by visually impaired and blind people. The improvement is worth and as the advantage of generating relatively precise trajectories when compared to approximate trajectories of the cases without assistance. We also observed that users with assistance perform even better when they have a previous knowledge of the environment. Most of the users indicate that location instructions were relatively precise, especially when direction instructions were given when they were located in the corners of the stations (location in a corner is of interest as this represents the case of a user entering in a *Metrobus* station). In some cases, the instructions were not precise and rather approximations. In this case, the system requires a second request to give a more precise instruction.

Overall, the main functionalities of our experimental setup can be summarized as follows:

- The mobile assistant locates exit(s) in appropriate form in most of the cases. The errors are mainly caused by the imprecision of the GPS and compass device.
- The system identifies at what direction are located the boarding gates (left, right, front or back) with a small error of direction according to 50 samples made by blind and visually-impaired people in the development and experiment process.
- Browsing audible menus have been considered as easy and the audio understandable, according to the group blind and visually-impaired users that have tested the system. Navigation keys (key # 5 to accept; key # 2 to up option # 8 to down option) are easy to locate (cell phones have a mark on the key number 5 to facilitate manipulations).
- The system identifies the station inside and outside it about 20 meters around. This limitation is due to the characteristics of the compass device used in the experiments.
- At the limits of the station, it appears that direction errors increase. This can be solved using additional points of reference with corresponding cardinal directions. In other words, at the limits of stations additional control points are required to improve the precision of the values obtained from the compass device.

Overall, the user evaluation of the mobile assistant gives them a better sense of security and confidence during their behavior, this also contributing to their overall navigation performance. Indeed the mobile assistant developed so far does not provide a perfect system as GPS positioning errors can still happen depending on the configuration of the GPS satellites, and as, for example, the traffic conditions of a given station can hamper navigation at busy times. Regarding phone consumption, Bluetooth communications between the Cell phone and the GPS and compass device hampered the battery autonomy by 50% (i.e., 12 to 6 hours), this being still acceptable when considering the service delivered to the blind people. Current experiments involve development of the whole application concept within an Iphone 4 device integrated with a GPS digital compass.

5 Conclusion

Navigation systems have been widely developed other the past few years. Their range of applications is still increasing, from indoor to outdoor environments. This is why the functionalities of these systems should be not limited to common users, but also extended to communities with disabilities. The research presented in this paper introduces an experimental audio-based system that facilitates navigation of blind users in the *Metrobus* of the city of Mexico. The system is based on a cell-phone setup extended by a GPS and a compass device. The functionalities developed so far give audio-based instructions to locate boarding gates and exits in the *Metrobus* stations. Preliminary experiments are encouraging and show that blind people find the main principles of the system developed satisfactory and are worth further developments. Overall the approach and the system developed so far provides an audible interface for blind people that has been and which is still experimented in the *Metrobus* system of the city of Mexico. The whole system is still under development; several experiments are under progress and applied to other *Metrobus* lines.

Further work is planned at the design and implementation levels. For example, additional modules to search for routes within the *Metrobus* system and estimated time of travel, closing stations and others considerations can be explored, taking into account similar principles, that is, providing audio-based interfaces to blind users. Integration of daily and warning information are other directions to explore. We also plan to implement our design using novel generations of cell phones in order to improve the integration of the positioning and directional devices into a single one. At the implementation level, additional testing is still necessary to implement the approach to other cell phones, particularly the ones with low processor power. The whole approach can be applied to other environments, either outdoors or indoors, such as bus systems and subways. Overall, exploiting the capabilities that cell phone applications offer it can be possible to expand the telecommunications market covering a population somewhat marginalized as are the blind persons, and this considering social aspects and not only market profits.

Acknowledgements. The authors wish to thank the anonymous reviewers for their useful comments and the Mobile Computing Laboratory of the Instituto Politécnico Nacional in Mexico, IPN-UPIITA, COFAA-IPN, and SIP for their support.

References

1. Darvishy, A., Hutter, H.-P., Früh, P., Horvath, A., Berner, D.: Personal mobile assistant for air passengers with disabilities (PMA). In: Motion, - Working Papers in Transportation Systems. School of Engineering, vol. 2. ZAW, Winterthur (2008)
2. `http://www.datasheetcatalog.net` (last access October 2008)
3. Dybkjaer, L., Hemsen, H., Minker, W.: Evaluation of Text and Speech Systems. Text, Speech and Language Technology Series, vol. 37. Springer, Heidelberg (2008)
4. Golledge, R.G., Stimson, R.J.: Spatial Behaviour: a Geographic Perspective. The Guilford Press, New York (1997)

5. Heuten, W., Henze, N., Boll, S., Pielot, M.: Tactile wayfinder: a non-visual support system for wayfinding. In: Proceedings of the 5th Nordic Conference on Human-Computer Interaction: Building Bridges (NordiCHI 2008), Lund, Sweden, vol. 358, pp. 172–181. ACM Press, New York (2008)
6. Hub, A., Diepstraten, J., Ertl, T.: Design and development of an indoor navigation and object identification system for the blind. In: Proceedings of the 6th International ACM SIGACCESS Conference on Computers and Accessibility, Assets 2004, Atlanta, GA, USA, October 18-20, pp. 147–152. ACM, New York (2004)
7. Hub, A., Hartter, T., Ertl, T.: Interactive tracking of movable objects for the blind on the basis of environment models and perception-oriented object recognition methods. In: Proceedings of the ACM SIGACCESS Conference on Computers and Accessibility, Portland, Oregon, USA, pp. 111–118 (2006)
8. `http://www.inegi.org.mx/lib/buscador/` (last access: December 2008)
9. Jacobson, D.: Cognitive mapping without sight: four preliminary studies of spatial learning. Journal of Environment Psychology 18, 189–305 (1998)
10. Jacobson, D.: Representing spatial information through multimodal interfaces: overview and preliminary results in non-visual interfaces. In: Proceedings of the 6th International Conference on Information Visualisation, pp. 730–734. IEEE Press, Los Alamitos (2002)
11. Kitchin, R., Blades, M., Golledge, R.G.: Understanding spatial concepts at the regional scale without the use of vision. Progress in Human Geography 21(2), 225–242 (1997)
12. Klatzy, R.L., Lippa, Y., Loomis, J.M., Golledge, R.G.: Encoding, learning and spatial learning of multiple object locations specified by 3D sound, spatial language and vision. Experimental Brain Research 149, 48–61 (2003)
13. Na, J.: The blind interactive guide system using RFID-based indoor positioning System. In: Miesenberger, K., Klaus, J., Zagler, W.L., Karshmer, A.I. (eds.) ICCHP 2006. LNCS, vol. 4061, pp. 1298–1305. Springer, Heidelberg (2006)
14. Rehrl, K., Bruntsch, S., Mentz, H.J.: Assisting multimodal travelers; design and prototypical implementation of personal travel companion. IEEE Transactions on Intelligent Transportation Systems 8(1), 31–42 (2007)
15. Rice, M., Jacobson, D., Golledge, R.G., Jones, D.: Considerations for haptic and auditory map interfaces. Cartography and Geographical Information Science 32(4) (2005)
16. Shinohara, K.: Designing assistive technology for blind users. In: Proceedings of the 8th International ACM SIGACCESS Conference on Computers and Accessibility, Assets 2006, Portland, Oregon, USA, October 23-25, pp. 293–294. ACM Press, New York (2006)
17. Virtanen, A., Koskinen, S.: Navigation System for the Visually Impaired based on Information Server Concept, VTT Technical Research Center (2010), `http://virtual.vtt.fi/noppa/materiaali.htm` (last access: October 2010)
18. Vollmer, J.: PAVIP-Feldversuch in Bern. SZB-Information, Fachzeitschrift für Sehbehindertenwesen 133, 45–46 (2004)
19. Willis, S., Helal, S.: RFID information grid and wearable computing solution to the problem of wayfinding for the blind user in a campus environment. In: Proceedings of the 9th IEEE International Conference on Wearable Computers, pp. 34–37. IEEE Press, Los Alamitos (2005)

A Rich Cloud Application to Improve Sustainable Mobility

Sergio Di Martino, Clemente Giorio, and Raffaele Galiero

University of Naples "Federico II", via Cinthia – I-80126 Napoli, Italy
dimartino@na.infn.it

Abstract. Improving the efficiency of public and private transportation systems is paramount to reduce pollution and improve quality of life. GIS and other IT solutions can support *Sustainable Mobility* in various ways, such as suggesting the most suited mean of transport for a given context, or optimizing the occupancy rate of vehicles by recommending other passengers interested in sharing a ride. In this paper we present a technological demonstrator for a Cloud computing-based platform, suited at favoring Sustainable Mobility. In particular, the proposed system helps users in choosing the most environmentally friendly transport solution, matching his/her needs, preferences, and actual location. From a technological point of view, the proposed platform takes advantage of the Cloud infrastructure, resulting in a *Software plus Services* solution. This choice is motivated to offer high reliability and to easily scale up as the number of users increase. Many different channels can be exploited to access the service, namely a Rich Internet Application, a Mobile application, or even a plug-in for in-car telematics systems, all based on the Microsoft Silverlight technology. Moreover, the platform also exploits a (geo)social network to improve the users' confidence in the rides arranged with other passengers. The described platform won the 2010 Italian finals for the Microsoft Imagine Cup student's contest.

Keywords: Sustainable Mobility, Telematics, LBS, GeoSocial Networks, XPS.

1 Introduction

According to the International Energy Agency, the global emissions of carbon dioxide (CO_2) released by the processes of fossil fuel combustion was 30 billion metric tons in 2007, with an increase of 3.2% on 2006 [4]. Among these, 23% of the CO_2 emissions refer to the transport sector that is also the main cause of oil dependency worldwide (59% of total oil consumption [4]). Clearly some countermeasures should be undertaken to reduce CO_2 emissions, as highlighted also by the United Nations within the "8 Millennium Development Goals" [16].

In this direction, the concept of *Sustainable Mobility* has been introduced to refer to any mean of transport with low impact on the environment. The most common types of Sustainable Mobility are Bike- and Car-sharing, i.e. rental from a shared fleet of bikes or cars respectively, public transport, and carpooling. In particular the latter is one of the most simple and effective enhancements that can be done on

K. Tanaka, P. Fröhlich, and K.-S. Kim (Eds.): W2GIS 2011, LNCS 6574, pp. 109–123, 2011.

environmentally friendly transport solutions. Indeed, carpool means improving the vehicles occupancy rate, that was only 1.06 persons per car in 2005 [15]. The rationale for the carpooling is that, if some travellers have similar departure and destination points, they could share vehicles, with significant benefits for themselves (reduced trip costs and stress) and for the environmental wellbeing (reduced number of cars on the road and of pollutant emissions).

More broadly, Sustainable Mobility can provide benefits in terms of:

- Reducing environmental impact,
- Decrease in traffic,
- Reducing moving costs,
- Promoting social relationships,
- Reducing stress due to driving.

However, despite many efforts and initiatives, to date the most of IT solutions to support Sustainable Mobility have reached only a partial success. Motivations are both technological and social. The challenging part with carpooling is to define a mechanism suitable to coordinate "on-the-fly" ride offerings and requests, based on users' positions [8] and preferences. A lot of web portals are arising, providing basic features such as the possibility to ask for a ride in a given date/time, but to date many appealing concepts, like using location awareness thanks to the GPS sensors in the smartphones, are not adequately exploited for improving the service. Among other technological motivations for the lack of diffusion, it is worth recalling that:

1. The long term scheduling of carpooling has been proven to be a NP-complete problem [9]. Some heuristics have been defined for constrained versions of the problem [1], but in any case if the community of users becomes wider, there are significant computational issues.
2. Data fusion among different data sources of the various means of transport may be challenging.
3. A platform working only on the internet will be very limited [15], being able to serve in real-time only a fraction of potential carpoolers. Integration within automotive on-board telematics systems could highly improve the diffusion of the service.

Also the social challenges are far from trivial, especially when dealing with internet-based solutions for car-pooling and -sharing. Among these, there is:

1. Lack of security for participants [8]. Sharing the car with unknown participants may discourage people to use these systems.
2. Participants may highly vary in terms of cultural, educational and financial backgrounds. People with very different profiles may feel uncomfortable together in a ride arranged only based on their location and trip details.
3. Lack of reliability for pre-arranged rides. Often those systems leave to the passengers the task to arrange the ride. So they do not provide location-based features to check if a ride is on time, or if there are any problems.

In this paper we describe the technological demonstrator we developed for a platform to face the Sustainable Mobility problem. The innovative idea of the proposal, named *Lift4U*[1], is to coherently integrate within a Cloud-based architecture (to overcome the computational issues), a wide set of sustainable means of transport (to increase the number of suggested solutions) with typical social network concepts (to overcome part of the social challenges). Finally, these services are accessible thanks to three different types of clients (desktop-based, mobile, and in-car telematics), with the goal to improve the capillarity of the information distribution and the quality of the provided location-based services. The back-end of the described platform has developed upon the Microsoft Azure solution, while the clients exploit the Microsoft Silverlight technology.

The paper is organized as follows. After a section presenting related work, the ideas behind the proposed approach are described in Section 3. Then, in Section 4, an overview of the employed technologies is provided, while in Section 5 the multichannel clients are described. Some final remarks and conclusion are also presented.

2 Related Work

In the literature (and on the market) there are many proposals and solutions falling in the domain of the Sustainable Mobility, but at the best of our knowledge, none of them is able to provide the set of services we are proposing.

Most of the existing works are vertical, addressing only one mean of transport (e.g.: compare different airfares or car rental rates) or a specific category of intended users with a repetitive transport pattern (e.g.: commuters, employees, students, etc… [8]). Among them, it is worth citing the online community of *NuRide* [12], where users earn rewards from sponsors as they use the system to arrange rides. Other remarkable proposals are *eRideShare* [4], and the Italian Highway agency proposal [1]. In any case, in our opinion an IT solution limited to a single mean of transport has two main drawbacks:

1) User is not able to effectively compare various transportation solutions and choose the most suitable for his/her aims.
2) User is not made aware of the existence of other Sustainable Mobility solutions, other the one supported by the specific solution.

About carpooling, there are various works addressing the problem from an algorithmic point of view. These works mainly aim at optimizing the selection of passengers and the route for their pick up. For instance, in [17] authors propose an optimization algorithm to reduce the idle time for carpoolers, while minimizing the number of vehicles on the road. However these works do not consider the social issues previously described.

---

[1] *Lift4U* was the winning proposal at the 2010 Italian finals of the Microsoft Imagine Cup student's contest [9]. Further details can be found, in Italian, at the site: http://www.lift4ucarpooling.com/

On the other hand, integration between carpooling and social networks in provided in *TravelRole* [12], but this solution does not exploit location-based services, neither provides support for mobile users.

Other interesting approaches are those presented in the domain of the Intelligent Transportation Systems, such as in [19]. However, to the best of our knowledge, none of these solutions tries to merge social network benefits with smarter route plannings.

Finally, many carmakers are moving toward Sustainable Mobility solutions, such as the Daimler *car2gether* system [4], but they handle only car sharing/pooling, and often are limited to small geographical areas.

3 Lift4U: The Proposed Platform

In this section we will describe the platform we are proposing, named *Lift4U*, in terms of provided services and features, while technical details are provided in the next one.

3.1 The Features Offered by Lift4U

As described in the previous section, nowadays there are available on the web several utilities to support users in arranging a trip. Nevertheless, usually these applications deal with a single modal choice. The result is that, in order to find out, compare, and choose the most viable plan, user has to spend a lot of time, gathering fragmentary information and then "squeeze out" the solution. Thus, in our opinion, to date there is a critical lack of centralized software applications able to compare various transportation solutions and propose the one that will fit better some given user's needs and location.

To overcome some of these issues, we propose Lift4U, a web-based platform to support users in arranging environmentally sustainable trips that best fit their profiles, by checking and comparing different transportation modalities.

The innovative idea is the provision of an integrated platform that embraces many modal choices with a low environmental impact, such as carpooling, car sharing, bike sharing, walking, mass transit, and taxi sharing, and is able to compute the ecological and economic impact of each solution. In particular, once departure and destination points have been specified, Lift4U gives as output a set of mobility solutions among those fitting user preferences, each one with estimated costs, commuting time and CO_2 emissions. Moreover, Lift4U is able to arrange carpools, where passengers are chosen taking into account also social networks, in order to mitigate social issues.

Many different channels can be adopted to access the platform, namely a web browser, a smartphone, or an in-car telematics system. Each one has its peculiarities, allowing to exploit a different range of services. For instance, a GPS-equipped smartphone will be useful to provision a real-time, location-based transportation solution, while a web browser will support user in specifying his/her preferences in a more usable fashion.

A high-level conceptual schema of the proposed platform is depicted in Figure 1. In particular, to get information about non-private transport solutions, the platform is able to interact with a set of as for the external data sources. The most of GIS services are handled by an external map service (in our case the combination of Microsoft Bing

Maps and MapPoint), while many different external social networks are involved to get information about users' connections. The platform has three main modules running on the Cloud, namely a handler able to dispatch requests and to compute the different solutions (the *Sustainable Mobility Module*), a *GIS Module* to limit the interaction towards the external data sources to only those geographically involved in a given trip, and finally a *Carpool Module* to arrange the pools of passengers. A remote database is used to store some information about users, while a set of different intended clients is made available to improve the spreading of the platform.

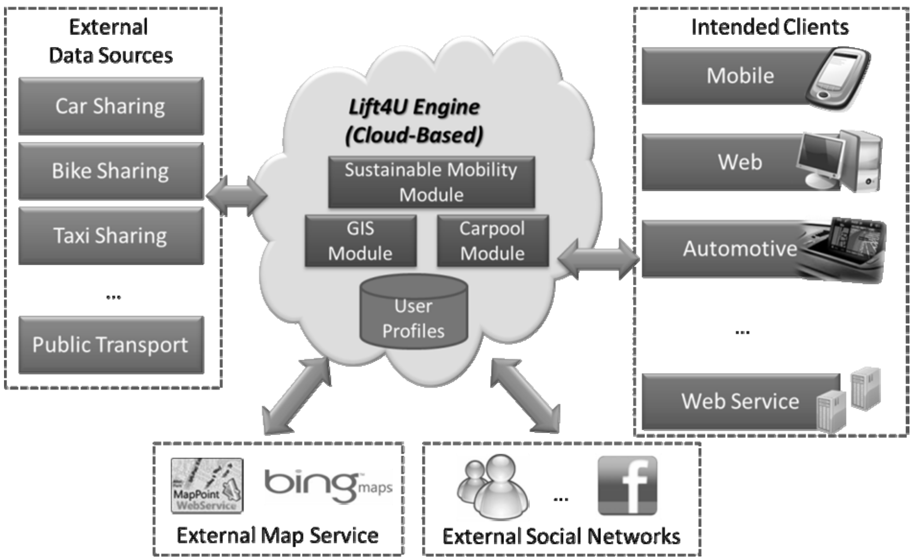


Fig. 1. Conceptual Schema of the Lift4U Platform

3.1.1 The GeoSocial Network

To date the spreading of car sharing and pooling is limited by some social issues. Probably the biggest one is the lack of trust in unknown people that could share a ride.

A key point of the Lift4U proposal to address this issue is to take leverage on the concept of "community" composed by its users. Indeed, in our opinion, exploiting the social links that will form in a community among users is fundamental to increase confidence in other users, by means of:

1. A friendship mechanism, where *friends,* or *friends of friends,* are preferred in the arrangement of a carpool or other mobility solutions that involve the sharing of a vehicle with other people;
2. A feedback mechanism, such as the one included in *eBay*, gauging the behavior of a user in past rides. Indeed, at the end of each trip, passengers can rate each other, and in case add also comments that will be valuable for future carpoolers. This concept holds also for other services, such as public transport, car sharing or bike-sharing, where users can leave ratings and comments on their satisfaction.

Since to date there are available on the web many global-coverage social networks with a huge number of users (Facebook for example has more than 400 million users), it makes no sense to define from the scratch a new community for our platform. Rather, we based the Lift4U community onto other very diffused social networks, to highly increase the potential catchment area of the platform and disseminate the benefits of sustainable mobility. Indeed, Lift4U can be connected to other social networks, such as Facebook, Twitter or MSN Live, to access users' profile information. As a consequence, in our case we can talk of a GeoSocial Network, since we take into account not only the user profile and preferences, but also his/her actual location, to provide more tailored, location-based services, described in the following.

3.1.2 The Location-Based Services

The platform can be exploited through two main different channels:

1. A Rich Internet Application, declined for both a nomadic or a stationary users, or
2. A web service, intended for any enterprise or public institution willing to provide sustainable mobility solutions within its own web portal.

Since both these solutions are based on a Cloud infrastructure, to overcome some computational issues, and to scale accordingly to the number of users, we can talk of a Rich Cloud Application (or RCA), and of a Software-as-a-Service (or SaaS). Details on these paradigms are provided in Section 4.

Focusing on the Rich Internet Application, to date it has been implemented in two different versions, namely a standard desktop application, and a lighter mobile solution, able to exploit the positioning sensors of newer mobile phones, to provide Location-based services.

The mobile client offers new interesting scenarios, leading to a conceptual shift from the old *travel planning* to a newer real-time dynamic arrangement of sustainable solution, optimizing users' mobility, by knowing his/her position and preferences. Indeed, in the case the transport solution is the carpool, the platform will automatically find "compatible" people, form both the spatial and the profile points of view. Once a user advices him/herself as a driver, he/she specifies the departure and destination points, the number of available seats, and a maximum allowed distance of detour to pick-up other passengers. Then, the platform creates an area of interest (or a *tunnel*) around the original route, and checks the list of waiting passengers, looking for all the departure and arrivals points falling within this tunnel. This allows identifying not only the passengers making the same trip, but also those who share only a part of the journey. Once all the involved people have agreed on the arranged trip, the mobile clients of the passengers will be notified of the carpool position, alerting their owners for the estimated pick-up time or for delays.

These Location-based services exploit the Hybrid Positioning Systems (XPS) technology that complements satellite navigation systems, such as the GPS, with Local Positioning Systems (LPS), exploiting other technologies to understand user position. In our case, we used a lookup mechanism on the IP of the client to infer its location. The combination of these technologies helps us in overcoming the limitations of common Assisted-GPS (A-GPS), limiting the "urban canyon effect" between tall buildings, and working also indoors with an acceptable precision in the calculation of position.

In the future we envision a third version of the client (which is under development), intended as a plug-in to be installed in Automotive Telematics Devices equipped with Microsoft Auto. In this way, a driver interested in creating a carpool could post the availability of seats on the Lift4U platform in a straightforward way, since some information are automatically inferred by the vehicle and in-car navigator status.

4 Technological Details on Lift4U

In this section we present the technological details underlying our proposal. In particular, we'll start by describing the employed architecture, and then we'll show some of the web and mobile user interfaces we have developed.

4.1 The Lift4U Architecture

The main design criteria we adopted in the definition of the Lift4U architecture were to get an agile application infrastructure able to:

1. Easily integrate newer sustainable solutions, as soon as they become available.
2. Scale up as the number of users increments.
3. Be reliable and highly available.

These non-functional requirements, together with the heavy computational needs of Lift4U's adaptive route matching algorithms, motivated us to implement the back-end on a Cloud Computing infrastructure.

Cloud Computing has been defined as "[…] *a model for enabling convenient, on-demand network access to a shared pool of configurable resource (e.g. networks, servers, storage, applications and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction*" [15]. Indeed, outsourcing IT commodities to a Cloud Computing provider brings many benefits. Among them, there is a higher availability, due to the high redundancy of computational power in the data centers, and an excellent scalability (the Cloud is usually based on a pay-per-use concept, where additional computational resources can be allocated on demand). Also from an ecological point of view, the Cloud solution can bring many benefits, since with this paradigm, different enterprises share computing power as well as data storage, with a drastic improvement of energy efficiency.

As for the Lift4U back-end, we exploited Microsoft Azure technology, that is a Platform-as-a-Service (PaaS) [15], providing a distributed operating system (mainly based on Windows 2008 Server) useful to develop custom Software-as-a-Service (SaaS) [15] on top. On the other hand, the necessity to provide Location Based Services, obtained by means of the Hybrid Positioning System (XPS), and in general an appealing user experience, requires that some logic runs on the client devices.

The *Software plus Services* (S+S) paradigm [15] meet these architectural needs, since it combines the appealing features of the server-side *SaaS* model with the advantages of software running on clients. Indeed, S+S can be considered as one of the most advanced architectural patterns, where the back-end is implemented by a Service Oriented Architecture (SOA) running in the Cloud to offer scalable computational power, and front-end is provided by a Rich Internet Application (RIA) [1], able to fully exploit the potentialities of newer client devices, such as positioning sensors, accelerometers, and so on.

When dealing with the Microsoft technology stack, an S+S system is known as *Rich Cloud Application* (RCA), where basically the back-end is based on Microsoft Azure, and the front-end on Silverlight, as shown in Figure 2.

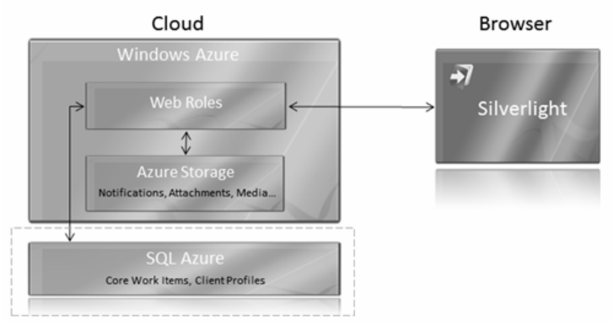


Fig. 2. The Rich Cloud Application framework, as proposed by Microsoft

As for the clients, they are based on the Microsoft Silverlight technology, which is a development platform proposed by Microsoft to offer interactive user experience. Among the features offered by this technology, one of the most appealing for our goals is the possibility to run Silverlight interface on Web, desktop, embedded, automotive and mobile applications, with a reduced effort due to the porting between these different platforms.

4.2 Integration with External Services

The integration with the external data sources, providing information on public transport was a challenging part. The main problem is that to date, to the best of our knowledge, there is no widely diffused standard to specify and exchange public transport information, such as timetables, status, and so on. In this direction, the definition of a new interchange standard could be of great utility to improve Sustainable Mobility.

Our solution to tackle this problem was the adoption of the *Adapter* Design Pattern [6], which allowed us to specify our own interface for the specification of public transport information. Then, we had to implement a wrapper for each available web service we found, to convert its actual signature into our defined interchange interface. An example is provided in Figure 3, where different web services providing car sharing facilities have been "adapted" to our own defined *ICarSharing* interface.

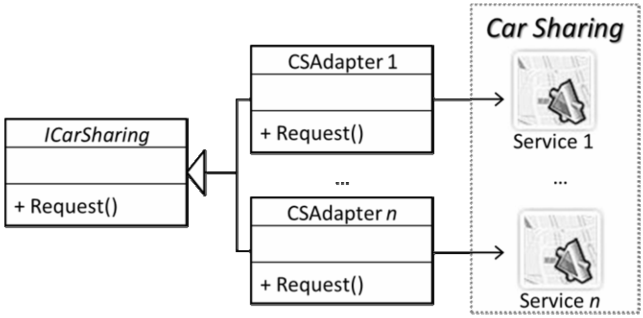


Fig. 3. Example of application of the *Adapter* design pattern

Moreover, for each provider, we keep track of the area covered by the service. For instance, we store the information that a specific bike-sharing service *BS1*covers the urban area of Rome, while *BS2* the area of Milan. This is a crucial information to reduce the number of providers to involve in computing a solution.

We acted in a similar way also to integrate other social networks that offer an API and allow communications over a REST XML interface, such as Facebook, Twitter or MSN Live. As an example, the integration with Facebook allows users to post on their walls, as well as to import profile information.

It is worth noting that thanks to the integration with other social networks, only application-dependent user information, such as preferences, feedbacks, comments, etc…, are stored in our database, based on Microsoft SQL Azure. In this way we keep track of a very small amount of information per user, leaving the most of his/her profile onto the main social networks' infrastructures. The traceability of a profile is handled by means of *OpenID* [13], a decentralized standard with over one billion enabled user accounts and over 50,000 websites accepting it. In this way a user does not need to create and maintain multiple credentials, and in the meantime it is accelerated the SignUp process. This brings also the advantage that a user's password is not stored in our platform.

Finally, the geospatial and route mapping services are orchestrated with the REST (Representational State Transfer) paradigm, described by WSDL 2.0 (Web Services Description Language). In the developed platform, we used the *BingMaps* and the *MapPoint* Web Services provided by Microsoft to manage the geographical information, to get information about route directions and route mapping, and to render maps onto the web clients.

4.3 How the System Works

In the following we will describe the two main scenarios of usage of our system, i.e. when the solution is based on public transport solutions, or when a carpool is arranged. These two different strategies require involving different modules and 3rd party services.

In Figure 4 there is depicted a simplified schema showing the modules involved in a trip based only on external mobility solutions. The process is triggered by the user, that specifies his/her departure and destination points though a client. This textual information is georeferred by querying an external map service provider, such as BingMaps. Thanks to this spatial information, the GIS module can discard all the external service providers whose coverage area is too far from the requested locations.

Then, each potential service is queried, to retrieve plausible transport solutions, which are merged and forwarded to the geospatial mapping service, now responsible to render them onto a map. Then, a ranking mechanism, based on user's profile, ecological and cost factors is applied to put in the highest positions the solutions that should best fit user preferences. Finally the core module will transmit the set of solutions to client that requested the service.

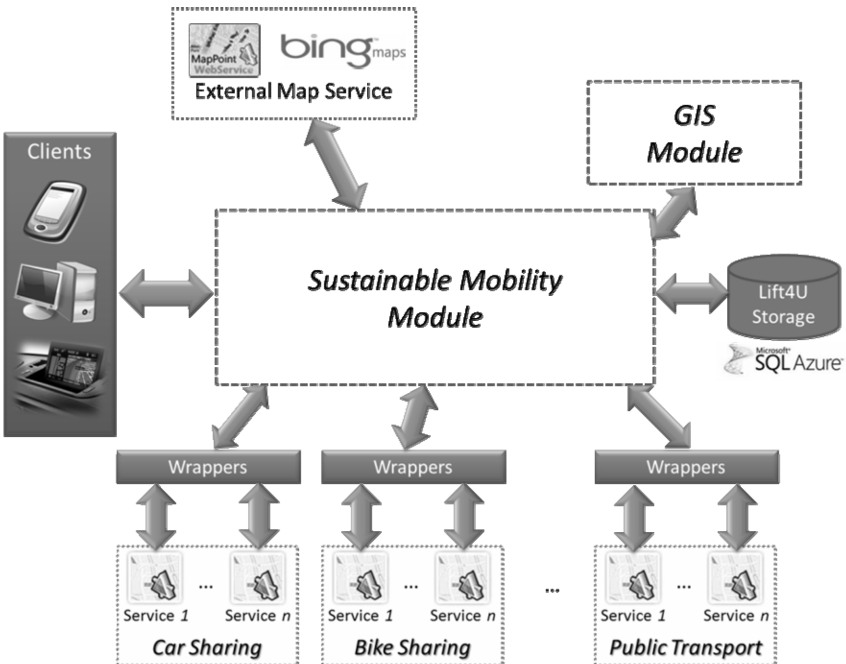


Fig. 4. Involved Modules with external mobility data sources

When dealing with carpooling, the schema is slightly different. Rather than querying external data sources, the platform looks for all the users waiting for a pick-up, whose path falls within the "tunnel" computed around the carpool driver's route. Then, a ranking mechanism promotes passengers that are in the driver's network of contacts, or those whose profile is most similar to the driver's one. Once the ride is arranged, information about status, pick-up time and location, and others are broadcasted to all the involved passenger clients.

5 The Multichannel Clients

In this section we will present the web and the mobile interfaces we developed for the Lift4U platform. We'll adopt a scenario-based description to describe the interfaces, showing a real example of use. User Bob lives in Milan and needs to travel to Warsaw for a business trip in a given date. Bob accesses the Lift4U website and inserts his credentials in a login page. Once recognized, he can specify his source and destination addresses, through the page shown in Figure 5.

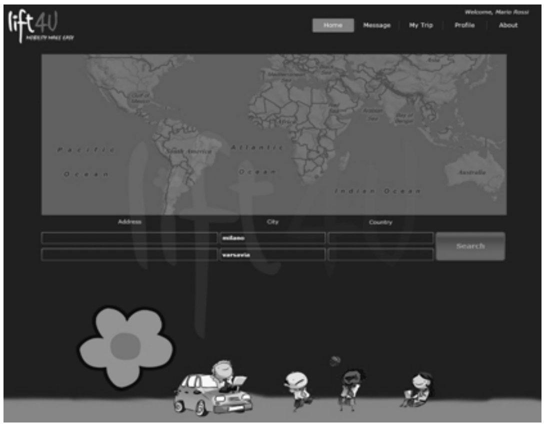


Fig. 5. The web page to specify departure and destination addresses

As described before, different sustainable means of transport will be considered by the system. As a first step, the system queries the map service, to get a total distance of the trip. This information is useful to prune some commuting modalities. For instance, since the distance between Milan and Warsaw is about 1,000 miles, of course walking and bike sharing are not considered. In particular, for such a long distance, only carpooling is proposed, as shown by the dialog in Figure 6.

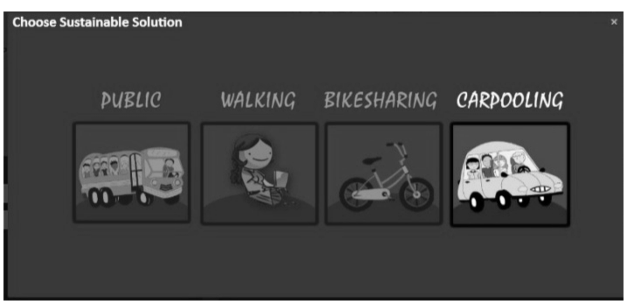


Fig. 6. Selection of the mean of transport

Once selected carpooling, Bob specifies he agrees to drive a carpool. In order to better match user's convenience, the system asks the maximum distance in miles Bob is willing to spend for "pick-up" detours, the number of available seats, an estimated starting period, and the type of vehicle fueling, to compute the trip costs and CO_2 emissions (Figure 7).

Fig. 7. Selection of the Date and maximum allowed detour

Once gathered all these information, Lift4U starts searching available traveling companions in the "tunnel" whose size is given by the defined maximum distance of detour. This is the most computationally expensive part of the process, achieved in the Cloud, by querying the Bing Maps route calculation service. Companions are searched initially in the Bob's network of friends. If no match is found, other potential passengers are identified, ranked by similarity between profiles. At the end of this step, Bob can browse the list of potential passengers on that route (Figure 8).

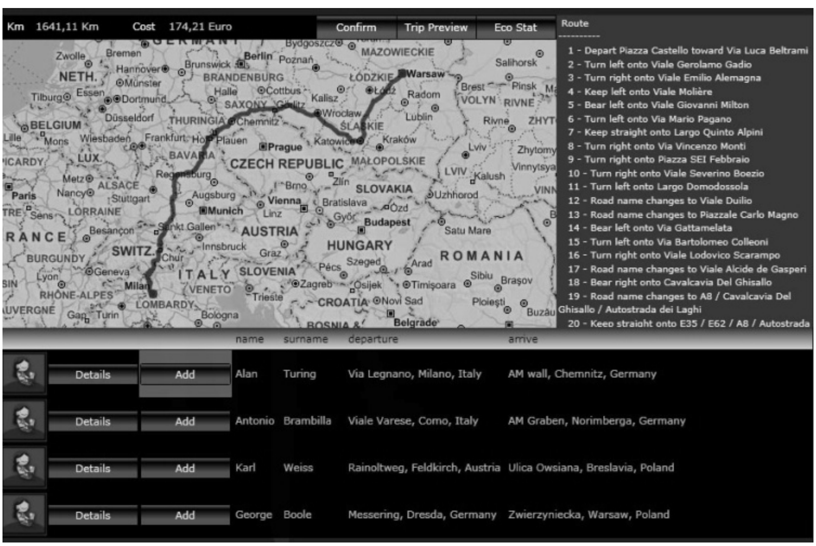


Fig. 8. Selection of passengers, and trip details

Bob can check their details, feedbacks and comments left by previous carpoolers who have already traveled with these users (Figure 9). Once he feels confident about the reliability of a passenger, Bob can add him/her to the carpool. The system updates the trip preview on the map, with the driving directions and the travel cost per head, to reach Bob's destination.

Fig. 9. Details on a potential carpooler

Then, the final route is previewed, its length with detours and cost effectiveness are displayed, and Bob can check the travel savings and the statistics about CO_2 reduction thanks to car-pooling, as shown in Figure 10 (left).

Fig. 10. The report page, showing costs and CO_2 emissions (left) and the Mobile Interface (right)

Alan and other selected passengers will receive a notification via the channel they have specified in their preferences, namely a text message, an e-mail, or an alert when activating the Lift4U mobile client. This latter case is shown in Figure 10 (right), where Alan receives a notification on his mobile device that he has been chosen by Bob for the Milan-Warsaw trip. Of course also Alan is free to confirm or reject the trip plan, viewing Bob's feedbacks and comments.

6 Conclusions

Reducing the ecological footprint due to transportation is one of the key goals to cut the emissions of greenhouse gases. Information science can heavily support this task, by suggesting to the users "smarter" mobility solutions among the various means of transportation.

In this paper we have described a new platform we developed, that tries to overcome the limitations of current sustainable mobility solutions, by coherently integrating a wide range of new technologies. Among them, the Microsoft Azure platform allowed us to develop a Cloud-based back-end, where the computation of route arrangements can be done exploiting the massive elaboration power of the Cloud infrastructure. As for the services, given a user's departure and destination addresses, together with other preferences, the proposed framework computes the transport solution with the lowest ecological footprint, considering various mobility alternatives, namely carpooling, car sharing, bike sharing, walking, mass transit, and taxi sharing.

As for carpooling, the use of a Geosocial network is included, in order to overcome some social issues that can arise when sharing a trip with unknown people. Indeed, the system tries to match users' profiles and preferences when combining a ride, taking into account locations and social links among users. The current location of the participants, obtained through Hybrid Positioning System, is taken into account to improve the affordability of the proposed arrangement.

Many other issues have to be still considered. For instance, as any community-based solution, there are strong issues about the privacy of the information, especially when dealing with tracking user movements. Moreover, to date the use of the Cloud helped us in overcoming problems in computing the most suited arrangement for carpooling, but newer heuristics should be defined, mixing coherently spatial and profile information to reduce the required computational resources. Finally, an on-the-field evaluation of the usability of the user interfaces will be shortly undertaken.

References

1. Allaire, J.: Macromedia Flash MX – A Next-Generation Rich Client, pp. 1–4 (2002)
2. Autostrade Carpooling, `http://www.autostradecarpooling.it/`
3. Coppersmith, D., Nowicki, T., Paleologo, G., Tresser, C., Wu, C.: The Optimality of the On-line greedy algorithm in carpool and chairman assignment problems, IBM Research Report (2005)
4. Daimler car2gether, `http://www.car2gether.com/`
5. eRideShare, `http://www.erideshare.com/`
6. Gamma, E., Helm, R., Johnson, R., Vlissides, J.: Design Patterns: Elements of Reusable Object-Oriented Software. Addison-Wesley, Reading (1994) ISBN 0-201-63361-2
7. International Energy Agency, CO2 Emissions from Fuel Combustion – Highlights 2009 Edition (2009), `http://www.iea.org/co2highlights/CO2highlights.pdf`

8. Lalos, P., Korres, A., Datsikas, C., Tombras, G., Peppas, K.: A Framework for Dynamic Car and Taxi Pools with the Use of Positioning Systems, computation world. In: Computation World: Future Computing, Service Computation, Cognitive, Adaptive, Content, Patterns, pp. 385–391 (2009)
9. Lift4U website, http://www.lift4ucarpooling.com/
10. Mell, P., Grance, T.: Draft NIST working definition of cloud computing, August 21, vol. 15 (2009)
11. Microsoft Imagine Cup website, http://www.imaginecup.com/
12. NuRide, http://www.nuride.com
13. OpenID, http://www.openid.net
14. Selker, T., Saphir, P.H.: TravelRole: A carpooling / physical social network creator. In: Proceedings of the 2010 International Symposium on Collaborative Technologies and Systems (CTS), pp. 629–634 (2010)
15. Steger-Vonmetz, D.C.: Improving modal choice and transport efficiency with the virtual ridesharing agency. In: Proceedings of Intelligent Transportation Systems, pp. 994–999. IEEE, Los Alamitos (2005)
16. United Nation Millenium Goal website, http://www.un.org/millenniumgoals/. (last visited on September 2010)
17. Vargas, M.A., Sefair, J., Walteros, J.L., Medaglia, A.L., Rivera, L.: Car pooling optimization: a case study in Strasbourg (France). In: Proceedings of the 2008 IEEE Systems and Information Engineering Design Symposium, pp. 89–94 (2008)
18. Varrentapp, K., et al.: The Long Term Car Pooling Problem – On the soundness of the problem formulation and proof of NP-completeness (November 2002)
19. Winter, S.: Intelligent Self-Organizing Transport. Künstliche Intelligenz (Special Issue on "Artificial Intelligence and Spatial Mobility") 3, 25–28 (2008)

Mobile Task Computing: Beyond Location-Based Services and EBooks

John Liagouris[1], Spiros Athanasiou[1], Alexandros Efentakis[2], Stefan Pfennigschmidt[3], Dieter Pfoser[1,2], Eleni Tsigka[3], and Agnès Voisard[3]

[1] Institute for the Management of Information Systems
Research Center Athena
G. Bakou 17, 11524 Athens, Greece
{liagos,spathan,pfoser}@imis.athena-innovation.gr
[2] RA Computer Technology Institute
Davaki 10, 11526 Athens, Greece
{efedakis,pfoser}@cti.gr
[3] Fraunhofer ISST
Steinplatz 2, 10623 Berlin, Germany
{stefan.pfennigschmidt,eleni.tsigka,agnes.voisard}@isst.fraunhofer.de

Abstract. Mobile devices are a promising platform for content delivery considering the (i) variety of attached sensors, (ii) widespread availability of wireless networks, (iii) even increasing screen estate and hardware specs. What has been missing so far is the adequate coupling of content to those devices and their users' actions. This is especially apparent in the area of Location-based Services (LBS), which, with few exceptions (e.g., navigation), have not fulfilled their predicted commercial success in mobile environments due to the following reasons: (i) content in typical LBS applications is still narrow and static, (ii) available methods and interfaces in mobile handsets for the discovery of available content are at best cumbersome (e.g., keyword-type search), and (iii) existing structured content available in LBS applications is hard to reuse. In this work, we propose the concept of task computing to complement and extend LBS as a means to enable the intuitive and efficient re-purpose, discovery, and delivery of rich content according to the user's needs. Further, we establish the theoretical foundations of task computing and its application in the LBS domain. We also present a fully functional prototype iPhone application structured around the concept of task computing.

Keywords: task computing, location-based services, ontologies, ebooks.

1 Introduction

Location-based services have emerged a few years ago to allow end-users to obtain information based on some location, usually the position of the user. Such services, for instance mechanisms to answer a query such as "Where is the nearest subway station?" or "What are the exhibitions in the city today?"

K. Tanaka, P. Fröhlich, and K.-S. Kim (Eds.): W2GIS 2011, LNCS 6574, pp. 124–141, 2011.

are currently receiving a great deal of interest. They manipulate the common aspects of location and time but also more complex notions such as the profile of a user.

General LBSs for mobile users are either extremely narrow and static in their content offerings, or address the needs of business users (e.g., fleet management). However, within the past few years, certain important arrivals of LBSs for everyday, casual users have emerged. These products and services focus either (i) on expanding the reach of geospatial information, or (ii) on coupling social network activities on a geospatial domain.

In particular, a class of services adequately represented by Google Maps and Google Earth for mobile devices, focuses on the simple task of delivering standard geospatial products (such as maps), along with "free text"-based search for web content. Other similar services come from Microsoft Bing Maps, Nokia Ovi Maps and so on. Though truly fascinating upon their arrival, these services have not changed the way users discover and consume content. Discovery still requires the users input from a hardware or on-screen keyboard. At best, a catalogue of available content is presented to the user, according to her search parameters and location. So the basic paradigm of how users manipulate geospatial information has not been altered. Rather, it has been poorly ported to mobile devices, simply by re-purposing available interface and information retrieval concepts for smaller screens. For example, a simple query for desktop-based users, such as "Is there a nice restaurant within 1km from location X that costs less than 10 euro per person?" requires several searches, and the gradual filtering/refinement of web content from several sources (e.g., maps, guides, blogs, portals, social networks, etc.). Obviously, a mobile user does not have the time, nor adequate hardware resources to perform this search.

The second class of products and services focuses on overlaying a geospatial mantle upon social networks and interactions. Users can annotate content with location attributes, whether it concerns their own interaction (e.g., geo-tag profile messages) or points of interest (e.g., leave a restaurant review). Such services include Facebook, Twitter, Yelp, Foursquare, Layar, and Google Buzz. They have also been embraced by the users with great interest, but they still remain a shallow means for general geospatial content discovery and delivery. While users can see where their friends are, read and submit reviews on POIs, etc., the prevalent paradigm we discussed earlier is kept intact. Moreover, these services highlight the problem even further, since they essentially comprise isolated islands of geospatial knowledge. For example, unless one has a Facebook account, she cannot search through the available content.

Stemming from the previous observations, we introduce the principles of task computing as a viable alternative to the problems we described. The contributions of this work include the theoretical foundation of task computing in mobile environments as well as the development of a complete framework containing all the necessary tools, libraries, and APIs to provide modularity and simple integration with existing solutions. Before continuing, we emphasize that our approach is orthogonal to more traditional tourism applications which recommend sites based on people profiles (with a recommendation system that can be more or less elaborate, see for instance [12]). In addition, it complements previous works based on the notion of task computing in that it addresses the problem of delivering rich fine-grained content to the users. In a nutshell, the main focus of our work is the *dynamic discovery, delivery, and presentation of rich and personalized content to mobile users based on the tasks they want to perform.*

The remaining document is organized as follows. Section 2 describes the concepts of task computing. Section 3 presents the principles of our approach and the overall architecture of the implemented infrastructure. Section 4 presents the proof of concept through the TALOS[1] prototype. We conclude in Section 5 with directions for future work.

2 Task Computing

It is a common sense that users organize their everyday lives around solving problems (tasks) and thus both services and content should be structured around tasks in order for them to be easily discovered and assimilated. This idea is strongly encouraged by the enlightening evaluation of NTT DoCoMo's task-based approach [18]. It is shown that the percentage of users reached the appropriate services by employing a keyword-type search through their handsets was no greater than 16%, whereas in the existence of a task-oriented search interface the corresponding percentage grew up to about 63%. According to the same test, it is also astonishing that 50% of the latter (one out of two) reached the services within five minutes, compared to just 10% (one out of ten) of the keyword-type search users.

Task computing [13,15,14] is a relatively novel concept in regards to the design, implementation, and operation of computing environments, aiming to fill in the gap between tasks (i.e., what users want to do) and services (i.e., functionalities available to the user). In contrast to the current traditional computing paradigm, task-oriented computing environments are ideal for non-expert users, since they provide access to useful information in a goal-centric manner, requiring minimal user adaptation to the particularities of the user interface and device characteristics. Furthermore, task computing is ideal for pervasive and ubiquitous environments, i.e., computing applications aiming to help users in accomplishing their daily goals. In such environments, users expose high demands for minimal interaction and show limited tolerance to ineffectual content. The organization of content and services around tasks offered in this discipline, has the potential to greatly improve the computing experience, since users receive timely and accurate information for the exact task in hand.

As mentioned in the previous, although the research in this area is still in its early stages, there are systems and prototype applications which demonstrate persuasive evidence about the importance and benefits of task computing. Besides the system developed by NTT DoCoMo labs which includes (i) a knowledge

[1] http://www.talos.cti.gr/

base of tasks that a mobile user performs in daily life, and (ii) a database of services that can be used to accomplish these tasks, another important work in this field is the Task Computing Environment (TCE) [20], a service discovery and composition framework for pervasive computing environments. TCE can reduce the cost of performing a task by displaying the tasks that could be performed by executing pervasive services located around the user.

2.1 Ontologies in a Task-Oriented Environment

Precondition for providing automated task-based services is the formal description of the potential tasks. This procedure amounts to the construction of a task model or, in other words, a so-called *task ontology*. Up to now, task ontologies have been used in various fields ranging from Artificial Intelligence [25,26] and Expert Systems [17] to Geographical Information Systems [23,9,10] and UI Modeling [22]. In general, the reason for their popularity lies in that they provide a flexible way for representing problem solving procedures, mainly because they facilitate sharing and reuse of knowledge along with automated reasoning capabilities [27].

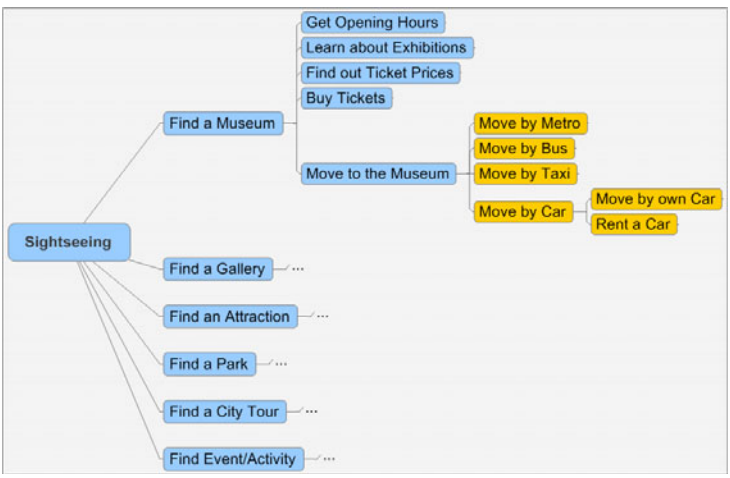


Fig. 1. Task Hierarchy Example

In the context of the Semantic Web [8], the definition of *ontology* could be that it is “a formal specification of a conceptualization”. Each ontology of the Semantic Web consists of two parts: (i) a vocabulary (intentional knowledge) that consists of concepts (classes) and relationships (properties or roles), and (ii) additional knowledge (instantiation) consisting of individuals, class and property assertions. A class assertion denotes that a specific individual belongs to a class, while a property assertion assigns a pair of individuals to a specific property. Ontologies in this context are used to provide order for a set of concepts and thus they are also referred to as *domain ontologies*.

Similarly, task ontology is a term referring to a formal model of tasks. According to the domain of interest, a task may represent a software procedure (e.g., "Sort an array of integers"), a business process (e.g., "Review the proposals") or even a simple human activity (e.g., "Cook food"). In our scenario, each task ontology includes the specification of the task attributes and parameters (e.g., name, input, necessary and/or sufficient conditions for accomplishing a task) and the definition of the relations between different tasks, such as subsumption and temporal ordering.

Intuitively, building a task ontology in our case amounts to modeling what the user of a mobile handset may want to do, e.g. "Go to the Theater", "Visit a Museum" or "Eat at a Restaurant". The basic feature of such an ontology is that complex tasks like those mentioned before are broken, following a natural human-like thought process, into simpler subtasks as shown in Fig. 1. Here, the hierarchy defined in the task ontology serves as a task-oriented index that is used for retrieving the appropriate content while guiding users to perform a task.

As shown in Fig. 2, users' tasks are described in the form of task ontologies whereas domain and *context ontologies* contain some of the data needed for instantiating these tasks, i.e., for capturing specific users activities.

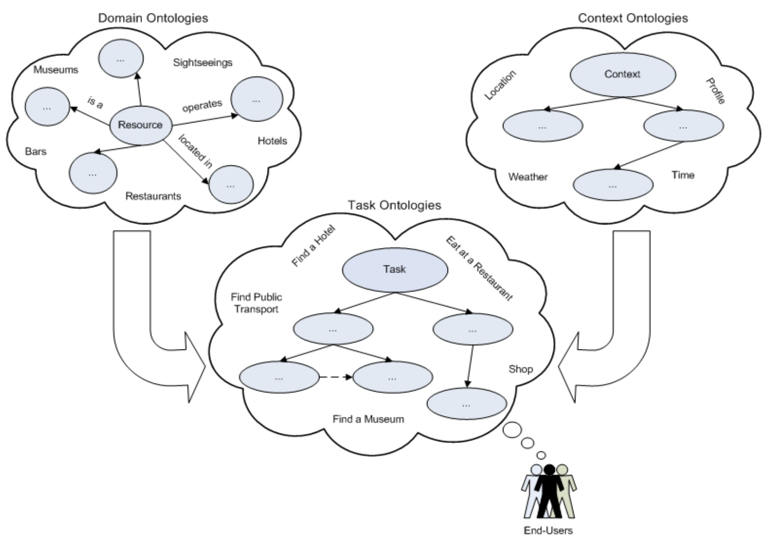


Fig. 2. Ontologies in a task-based service provision system

When defining a task, we may need to refer to one or more domain ontologies for concepts and definitions that are used to describe inputs, outputs, preconditions of the task and so on. Consider for instance the task "Find a Museum".

Here, a necessary list of museums in the area can be generated from the instances of a respective domain ontology. In general, by exploiting the expressiveness and flexibility of ontologies, we are able to perform the following:

- **Classification of resources:** Instead of just providing to the user a list of museums in the city that may prohibitively grow huge, especially for a screen of small size, the classification of museums according to their type can be very useful and time-saving when searching for the appropriate one.
- **Reuse of knowledge:** Already described knowledge about solving a task must be accessible for reuse in another task if suitable. Thus, by using ontology-based techniques, we are able to extract parts of solutions related to different tasks and combine them in creating solutions for new tasks. A representative example of this case is the knowledge about transportation as shown in Fig. 1. In this example, the highlighted group of tasks can be reused in more than one levels of the task ontology enabling the authors to easily define multiple "paths" for accessing the same resources depending on the tasks a user selects to perform.
- **Context-aware filtering:** The resources for accomplishing a task alter dynamically according to the current context, e.g., the user's location. Thus, by also modeling context, we are able to use conjunctive query answering techniques (including both context and structured content) for the efficient extraction of the most appropriate resources. A similar approach is followed in [19].
- **Access to various data sources:** By using the W3C standard languages for describing ontologies, i.e. the Resource Description Framework (RDF) [3] and the Web Ontology Language (OWL 2) [5], we can import and process data from various data sources existing in the web (e.g. DBpedia [1]). This can be done either in a forward-chaining manner, where data are processed and stored in the system database offline, or even directly through the mobile device when the user uses the application. One of the first attempts to store and efficiently manage voluminous RDF data in a mobile device with limited capabilities is presented in [24] with very promising results.
- **Consistency check:** When constructing a task ontology one has to pay attention not only to the syntax, but also to the semantics of the latter. The former, i.e. the syntax validation, can be ensured by restricting authors to construct the ontology through a graphical interface and automatically interpret the graphical notations into XML. However, regardless the correct syntax, there may be declarations in the ontology that contradict one another (e.g. two tasks each one of them requiring the other to have been performed previously). From our experience, allowing IT-illiterate authors to arbitrarily define their own task ontologies will definitely result in various logical contradictions or redundancies whose identification requires additional algorithms.

When faced with a particular type of content one has to be aware that not all content can be modeled using ontologies. For instance, the travel guide content that we use in our prototype is in the form of unstructured text. However, besides the part of the content, such as the POI-related data (e.g., museums, parks, etc.), which can be modeled using ontologies, unstructured text that is stored and retrieved into/from a relational database can still be used in instantiating the tasks of a task ontology. We address this issue in Section 4.3. At this point we emphasize that, in an ontology-based approach, POIs along with their properties (e.g., addresses, operating hours, specific features etc.) are regarded as pieces of well-structured information that is extracted from the overall available (unstructured) content and modeled using ontologies. This kind of POI-related information can be either static (extracted from a book) or dynamic material (retrieved from the web). Context-related information (e.g., date, time, location, weather, users profile) can be modeled using context ontologies [11] and, similarly to the case of POIs, it can be either static (e.g., the users profile) or dynamic (e.g., the location of the user retrieved on-the-fly).

2.2 Task Modeling

A task reflects what an end-user wants to do in a high-level layer of abstraction, e.g., "Visit a Museum". Each task is accompanied by a set of properties (e.g., input, output, precondition, etc.) and it is instantiated by (i) context and (ii) content in order to become an *activity*. Following the object-oriented programming paradigm, a task can be regarded as a class of activities (instances) that share common types of attributes. For example, if we assume that the task "Visit a Museum" has the attributes "Museum" and "Date" defined as inputs, then it can produce activities like "Visit the Museum of Acropolis on 26/6/10" or "Visit the Louvre on 28/7/10" and so forth.

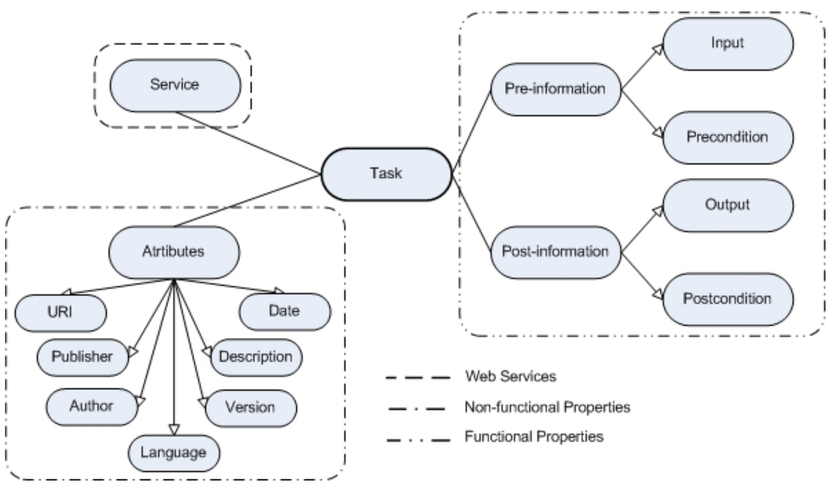


Fig. 3. Task Properties

Definition 1. *A Task T is a collection of well-defined attributes (or properties) P_i. T=P_i.*

As shown in Fig. 3, each property P_i of a task is classified under one of the following categories: (i) *Non-functional Properties*, (ii) *Functional Properties*, and (iii) *Services*. The first includes the metadata of a task (e.g., name, description, version, etc.) that are useful for the task authors. The second class includes all parameters taken into account when a task is performed by the user. This class is divided into two subclasses, (i) *Pre-information* and (ii) *Post-information*, that are described in the following. Finally, each task in a task ontology may be realized by one or more web services, e.g., the task "Book a Hotel Room" that is realized by a web service provided at "www.booking.com". In our prototype, the URLs of the respective web services are assigned by the author a priori.

Pre-information includes all parameters needed in order for a task to be executed successfully. This class breaks down into two disjoint subclasses: (i) *Input* and (ii) *Precondition*. An input stands for a parameter needed to perform a task and it can be defined as optional. For instance, the task "Find a Restaurant" may take as alternative inputs the exact location of the user in the form of longitude and latitude, as long as an abstract location in the form of the city or the neighbourhood name she is located in. In our model, task inputs can be (i) context-related parameters, e.g., the user's location, and (ii) POI-related parameters, e.g. the id of the specified mall in the task "Find Shops in the Mall" as retrieved from the user's local database (cf. Section 3). Note that, when defining a task, the author can specify groups of inputs that must be instantiated all together in order for the task to be performed.

Definition 2. *An input I of a task is a class defined along a hierarchy of concepts that belongs either in a domain or in a context ontology.*

On the other hand, a precondition represents conditions on the task inputs that must hold in order for a task to be performed successfully. Preconditions are logical expressions applying to (i) context, and (ii) POI-related parameters that exist in the user's local database. Regarding the task "Visit an Open Market", a representative example of the first case is "Weather = Sunny". In the second case, the precondition "Hotel.Rank = 5" (with Hotel defined as an input of the respective task) can be used in filtering the hotels retrieved from the user's local database when looking for a luxurious one. In our prototype, such preconditions are specified either a priori by the authors or on-the-fly by the end-user in order to act as an (optional) filter when searching for the most appropriate resources or services. From the UI perspective, they indicate the existence of a screen where the user can specify her preferences on the available task inputs and/or POI attributes.

Definition 3. *A precondition P_I of a task defines a restriction in the instantiation of a task input.*

Post-information includes all parameters that are generated after executing the task. It breaks down into two disjoint subclasses: (i) *Output* and (ii) *Postcondition*. An output describes information returned after performing a task. Similarly to the case of inputs, task outputs are classified under (i) context-related parameters, e.g. type of weather produced from the task "Get Weather Forecast" and (ii) POI-related parameters, e.g. the name and type of a POI that matches the specified input parameters. The difference with respect to inputs is that a task can produce as output a piece of content in the form of unstructured text that is retrieved either from a repository (static) or from the web (dynamic). Note that an output of a task (except the unstructured content) may be used as input in other tasks of the ontology introducing a dataflow. In our prototype, the default output of a task is the unstructured content retrieved from the user's local database (if available).

Definition 4. *An output O of a task is a piece of unstructured text or a class defined along a hierarchy of concepts that belongs either in a domain or in a context ontology.*

A postcondition represents conditions that must hold after performing a task. Similarly to preconditions, postconditions are logical expressions applying to (i) context and (ii) POI-related parameters both defined as outputs of a task. An example of a postcondition regarding the task "Move to the Park" could be something like "User.Location = Park.Location" where the "User.Location" is a context-related parameter referring to the user's position (defined as output of the task) while "Park.Location" is the location of the POI, i.e., the specified park, also defined as an output of the task. From the mobile application perspective, such a postcondition is useful for recommending tasks to the user and also for helping the user (re-)organize the schedule of tasks to perform during the trip.

Definition 5. *A postcondition P_O of a task defines a restriction in the instantiation of a task output.*

Before continuing, we point out that each input and output of a task in our prototype is instantiated by one of the following modules (cf. Section 3): (i) the *Context Aggregator* that manages all context-related attributes, (ii) the user (through the UI), and (iii) the users local database.

As far as task relations are concerned, we define four kinds of relationships between tasks: (i) *SubTaskOf*, (ii) *OR*, (iii) *CHOICE*, and (iv) *Sequence* relation. The SubTaskOf relation introduces a task hierarchy and denotes that the parent task is accomplished by accomplishing all of its children in any order. The OR and CHOICE relations introduce a task hierarchy as well, but in the former case the parent task is accomplished by at least one of its children, while in the latter case the parent task is accomplished by exactly one of its children (exclusive option). Finally, a Sequence relation denotes that a task is performed always before another one and it is accompanied by at least one parameter passing (binding) from the first task to the second one. For instance, the tasks "Find a Hotel" and "Learn about Hotel Facilities" are representative examples of this case considering that the second task always needs as input the output of the first one, i.e., the specified hotel. From the application perspective, different types of relationships between tasks in the ontology layer define different functional properties of the user interface (display order, redirection etc.).

3 System Architecture

The architecture of the overall system is illustrated in Fig. 4. The system has a typical three-tier architecture composed of (i) the *Data Sources* tier where the task ontologies and the available content are stored, (ii) the *Services* tier that includes all services implementing the business logic, and (iii) various *Clients*. In addition, two kinds of tools, the Task Ontology Authoring Tool (cf. Section 4.2) and the Content Authoring Tool (cf. Section 4.3), are provided to serve as a means to import, edit, and update the data sources. For the ease of presentation, in the remaining document we address the use case of a (mobile) traveler but the overall approach can be easily applied to other scenarios as well where task computing plays a central role.

A *Task Author* (TA) is responsible for creating the task ontologies based on the model we described in Section 2.2. By the time a task ontology is created and uploaded on the server, it is regarded as an *Idle Task Ontology* (ITO). This means that the ontology cannot be downloaded and used by the end-users until content is assigned to its tasks. The latter is performed by the *Content Author* (CA). By the time a CA assigns content to the tasks of an ITO, then the task ontology becomes an *Operational Task Ontology* (OTO) which means that it can now be downloaded (along with the assigned content) and used by the end-users. Obviously, besides the assignment of (structured and unstructured) content to tasks, the CA is also responsible for the overall management of content (import, edit, update, geocode etc.).

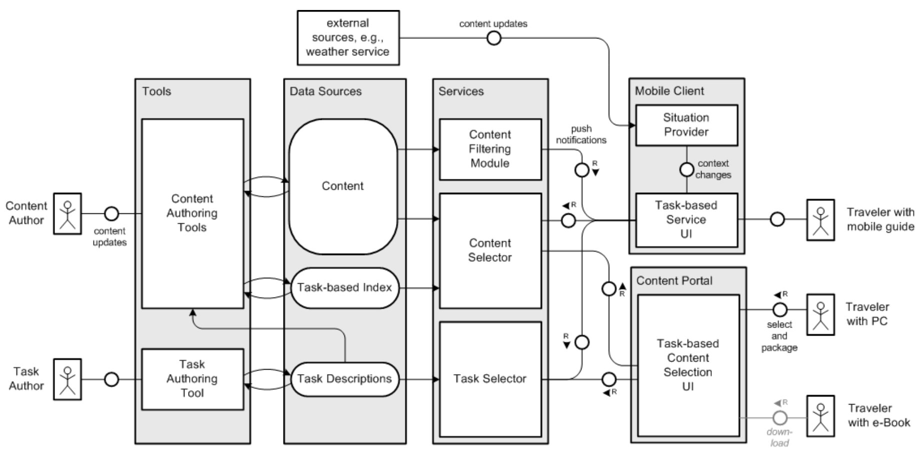


Fig. 4. System Architecture

The *Data Sources* tier encompasses (i) the *Task Knowledge Base* (TKB) and (ii) the *Content Base* (CB). TKB keeps all versions of every task ontology. The main reason we keep all versions of task ontologies in TKB is because, when downloading an OTO, the user of the mobile device may decide to include only a part of the assigned content and not all of it. In such a case, the rest of the content can be downloaded gradually during the use of the application. Thus, a downloaded OTO cannot be overwritten in the server until all end-users have updated their local copies; otherwise, in case the downloaded OTO is out of date, incompatibility problems may occur. Apart from this, a version history of the authors work can be very useful when, for some reason, one needs to rollback to an older version.

All available content is stored in the *Content Base*. More specifically, CB includes (i) unstructured content in the form of text, (ii) geo-referenced (structured) content in the form of POIs and their attributes, and (iii) the ids of the (operational) tasks to which the content is assigned (*Task-based Index*). As mentioned in the previous sections, POI-related data are described using an ontological model, hence, they are stored in CB following a variation of the widely-adopted database representations presented in [21]. As shown in Fig. 4, the dynamic content retrieved from external sources (e.g., web pages) is accessed by clients directly.

The *Services* tier encompasses components for recommending (i) tasks with respect to a user's *situation* as defined in [16] (*Task Selector*), and (ii) content from the underlying *Content Base* using a task selected by the user along with the user's situation as parameters (*Content Selector*). Task and content selectors are used when the user plans her trip (e.g., recommend content targeted to a backpack traveler), but also during the trip when, for example, an event occurs and an automatic re-scheduling is needed. However, the recommendation of tasks can also be done offline by using the context-related task parameters and preconditions defined in the operational task ontology. For example, in case the task author has specified a weather precondition for the task "Have a Boat Tour", then this precondition will be evaluated using the current situation of the user and in case there is a weather conflict, the task will be excluded from the recommended ones. At this point we emphasize that, although the task and content selectors support the use of content in a pull mode, a *Content Filtering Module* is used for notifying the user about just-in-time content updates in the server (push mode).

Regarding the *Clients* tier, there are two different clients that use the services of the system. First, a *Content Portal* facilitates the pre-selection of content based on tasks planned by the user (*Planning Mode*). In this case, the content can be compiled into a personalized travel guide eBook augmented with (i) additional information from the web, and (ii) a task-based index. Second, the content can also be compiled into a database (e.g., an SQLite file) that is downloaded and used as an offline source of content in the mobile device. In the latter case, the *user's local database* contains a (personalized) portion of the content stored in the *Content Base* of the central server that is coupled with the tasks of the downloaded task ontology. To access this information in a task-based and situation-dependent manner while traveling (*Trip Mode*), besides the actual content and the task-based index, a situation provider component (the

Context Aggregator) is included in the application. The situation provider uses various techniques to collect and aggregate context data in a pull-mode or even anticipate user situations. Context changes are forwarded to the client logic that calls the services and adapts the task and content recommendations accordingly.

4 The TALOS Prototype

This section describes the TALOS prototype infrastructure. The core objective in developing the prototype was not only to facilitate the end-user interaction through an intuitive and easy-to-use interface, but also to make task and content authoring as easy as possible.

4.1 General Information

The mobile travel guide provides a task-based interface where a mobile user can select tasks from a predefined task hierarchy and access relevant content. The *Context Aggregator* component gathers all data describing the user's situation, i.e., the user's current location, time, weather, and the user's traveler profile. Depending on the user's interaction with the UI, and the current context, content and tasks are recommended to the user, allowing for personalized information provision. Apart from accessing information, the user can also use the planning functionality supported by the mobile guide in order to create a trip schedule.

The available content in our prototype involves static content that consists of an existing travel guide for Brussels and a piece of geo-referenced content created by Michael Müller Verlag[2], as well as dynamic information from the web. Dynamic information includes (i) map tiles provided by the OpenStreetMap project, (ii) scraped information from various web sites concerning POIs (e.g., hotel sites), and (iii) information obtained by different web services, e.g., the Yahoo [6] and Geonames [2] weather services. Data storage on the mobile device is handled by an SQLite database [4]. A combination of the Apple Core Location Framework and wireless positioning techniques [7] are utilized in order to capture the users location (indoors and outdoors) and to offer context-aware content. The prototype is developed as a stand-alone application for the iPhone.

4.2 Task Authoring

The creation of task ontologies is done within the Task Ontology Authoring Tool (TOAT). TOAT provides an interactive 2D canvas where the authors can define the tasks of the ontology along with the relations among them by simply dragging elements from a palette. The result of this procedure is a Directed Acyclic Graph (DAG) where each node represents a task, and each edge represents a relation between two tasks. For the ease of presentation, the underlying task parameters (if defined according to the task model we described in Section 2.2) are not

[2] http://www.michael-mueller-verlag.de/

visualized in 2D, otherwise the graph would be quite difficult to handle. Instead, task properties are defined and shown from within a dynamic form shown in the right part of Fig. 5.

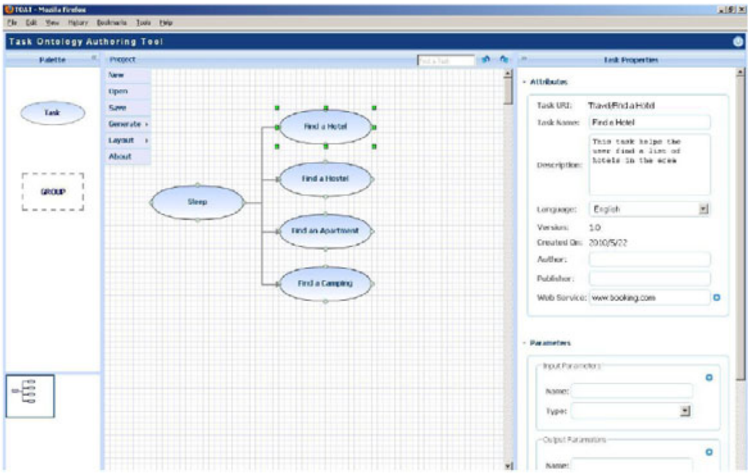


Fig. 5. Task Ontology Authoring Tool

After finishing the authoring procedure, the task author (TA) can get the XML file of the task ontology by clicking on the respective option in the menu. Before producing the XML representation of the ontology, TOAT performs a number of validations to ensure (i) that the file is valid according to a predefined XML Schema and (ii) that the ontology does not contain contradictions or redundancies. The XML file generated by the editor is stored in the central task knowledge base (TKB). This XML file defines the structure and the basic functionality (e.g., inputs, preconditions, dataflow, etc.) of the task-based UI. In this sense, it serves as an abstract model of the hierarchical user interface. The approach we follow here is reminiscent of the well-known Model-View-Controller (MVC) architecture paradigm where the view (what the user interacts with) is based on a generic model which changes (through the controller) according to (i) the user's actions, (ii) the current context, and (iii) the available content. Note that having a generic model as the basis from which the view is automatically generated provides us with great flexibility in managing the UIs, because the user's application can be easily updated by just changing the respective graph representations in TOAT.

4.3 Content Authoring

A core objective in the current effort was to make existing rich location-relevant content, in our case travel guides, readily accessible through the mobile device and the task-based interfaces. To do so we (i) linked such content to task metadata (task annotation), (ii) geocoded the content to provide a map-based access, and (iii) added dynamic web content to the mix by using web scraping, i.e., linking content from third-party sources to our content (e.g., linking the current exhibitions of a museum from the respective web page to the POI information stored in our content management system). The above tasks require a content management system, which in our case comprises a relational database (SQL Server 2008) and an interface.

Geocoding. To geocode the content, we relied on existing web services such as the Google Maps and Yahoo Maps API, Geonames, and Open- StreetMap name finder and developed an application wrapper that provides uniform access to any or all services depending on the user's needs and licensing restrictions. Advocating a semi-automatic geocoding approach, a map-based interface is introduced that allows the user (the content author - CA) to update the automatic geocoding results by dragging markers on the map. Fig. 6 shows the map interface.

Task Metadata. Using a task ontology as input, the content annotation tool provides a simple means of linking tasks to content. The content as stored in the content management system is shown in the web interface along with the task ontology represented in a tree structure. Clicking on a section highlights the section and shows a new pop up window with the suggested task hierarchy as a tree view and a list of already linked tasks with the specific section. After selecting a portion of the content, tasks are linked by selection to the selected content.

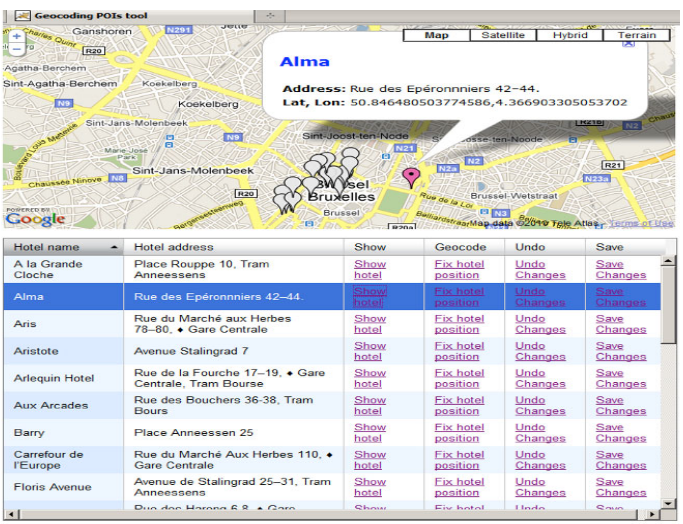


Fig. 6. Geocoding Interface

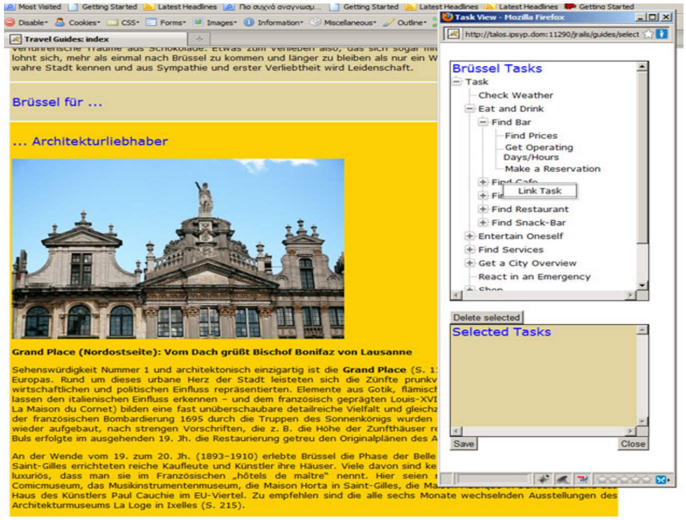


Fig. 7. Task Annotation Interface

Multiple selection is supported and the linked tasks are visualized accordingly. Fig. 7 shows the interface.

Dynamic Web Content. When considering electronic versions of print content, an important property is the possibility of frequent (and cheap) updates. To streamline publishing such content, we link third-party web content to our content base (CB). Examples here are opening hours and changing exhibitions of museums. Once such a link is established, it is simple to check if the information at the remote site has changed. We have developed web scraping tools, specifically a browser extension that allows one to mark content at a remote site and so to link dynamic web content to authored content. Fig. 8 showcases the tool.

4.4 iPhone Interface

The end-user interface of the mobile travel guide includes four different modes, namely the (a) *Activities*, (b) *eBook*, (c) *Map*, and (d) *Diary* mode as shown in Fig. 9. We point out that all these modes are interlinked to one another. Each one of them provides a different way to access available information and thus it presents another dimension of it.

The activities mode shows either the predefined task hierarchy or a context-adapted one. It offers a task-based UI which is generated dynamically, as defined in the task ontology. A task selection leads the user to appropriate content which can be a piece of unstructured text or a list of POIs. The map mode offers a spatial view consisting of a full screen map showing the (geocoded) POIs. Selecting a POI reveals content and task recommendations.

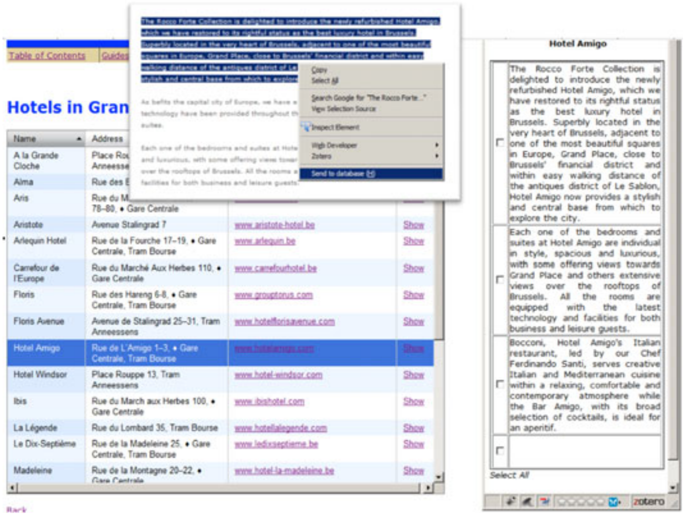


Fig. 8. Web Scraping Interface (plugin and web interface)

Fig. 9. iPhone screenshots. (a) Activities Mode: the task-based UI allowing for content provision and planning (b) EBook Mode: the content view offering structured and unstructured content (c) Map Mode: the spatial view showing POIs on a full-screen map (d) Diary Mode: the spatial view showing users plans and memories.

The eBook mode is the content view, where unstructured content, such as text derived from existing travel guides, and structured content, such as POI metadata can be read. For each unit of content (which could be a section of the guide or a POI) relevant tasks and POIs are suggested. Finally, the diary view offers a temporal view, where the user's plans and memories are presented.

The user selects tasks in order to plan her activities and stores bookmarks of all kinds of available content or personal pictures and notes to create trip memories, creating in this way a personal trip diary.

5 Conclusion

In this work, we investigated the potentials of dynamic discovery, delivery and presentation of rich content to mobile users based on the tasks they want to perform, an approach that leads to Mobile Task Computing. Starting from the theoretical foundation of the task computing paradigm in mobile environments, the contributions of this work also include a complete framework containing all the necessary programming tools, libraries, APIs, and authoring tools to provide modularity and simple integration with the existing solutions.

The core feature of our approach is an intuitive task model that can be used to describe the various activities of the end-users. All context-related parameters (user's situation) required for the dynamic discovery and personalization of the available content are integrated with this task model by exploiting the flexibility of ontological engineering.

Our future research directions in the field focus on collaborative task computing environments where users can share or recommend tasks to others and also on the combination of task and cloud computing techniques for retrieving resources and services available in the cloud by utilizing task-related knowledge.

Acknowledgments. This work was partially supported by the TALOS project, funded by the FP7 Research for SMEs work programme of the European Commission under contract number 222292. We would like to thank all partners in the TALOS project for their significant contributions in realizing this work.

References

1. Dbpedia, `http://dbpedia.org/About` (last accessed July 2010)
2. Geonames, `http://www.geonames.org/` (last accessed July 2010)
3. Resource description framework, `http://www.w3.org/TR/REC-rdf-syntax/` (last accessed July 2010)
4. Sqlite, `http://www.sqlite.org/` (last accessed July 2010)
5. Web ontology language, `http://www.w3.org/TR/owl2-overview/` (last accessed July 2010)
6. Yahoo weather, `http://weather.yahoo.com/` (last accessed July 2010)
7. Athanasiou, S., Georgantas, P., Gerakakis, G., Pfoser, D.: Utilizing Wireless Positioning as a Tracking Data Source. In: Proc. of the 11th International Symposium on Advances in Spatial and Temporal Databases, SSTD (2009)
8. Berners-Lee, T., Hendler, J., Lassila, O.: The Semantic Web. Scientific American 284(5), 34–43 (2001)
9. Kuhn, W.: Ontologies in Support of Activities in Geographical Space. International Journal of Geographical Information Science 15(7), 613–632 (2001)

10. Raubal, M., Kuhn, W.: Ontology-Based Task Simulation. Spatial Cognition and Computation 4(1), 15–37 (2004)
11. Bikakis, A., Patkos, T., Antonis, G., Plexousakis, D.: A Survey of Semantics-based Approaches for Context Reasoning in Ambient Intelligence. In: Proc. of the Artificial Intelligence Methods for Ambient Intelligence Workshop at the European Conference on Ambient Intelligence (2007)
12. Hinze, A., Voisard, A.: Location- and Time-based Information Delivery in Tourism. In: Hadzilacos, T., Manolopoulos, Y., Roddick, J., Theodoridis, Y. (eds.) SSTD 2003. LNCS, vol. 2750, pp. 489–507. Springer, Heidelberg (2003)
13. Masuoka, R., Labrou, Y., Song, Z.: Semantic Web and Ubiquitous Computing - Task Computing as an Example. AIS SIGSEMIS Bulletin, 21–24 (2004)
14. Masuoka, R., Parsia, B., Labrou, Y.: Task computing - The Semantic Web meets Pervasive Computing. In: Fensel, D., Sycara, K., Mylopoulos, J. (eds.) ISWC 2003. LNCS, vol. 2870, pp. 866–881. Springer, Heidelberg (2003)
15. Masuoka, R., Yuhara, M. (eds.): Task Computing - Filling the Gap between Tasks and Services. FUJITSU (2004)
16. Meissen, U., Pfennigschmidt, S., Voisard, A., Wahnfried, T.: Context- and situation-awareness in information logistics. In: Lindner, W., Fischer, F., Türker, C., Tzitzikas, Y., Vakali, A.I. (eds.) EDBT 2004. LNCS, vol. 3268, pp. 335–344. Springer, Heidelberg (2004)
17. Mizoguchi, R., Vanwelkenhuysen, J., Ikeda, M.: Task Ontology for Reuse of Problem-solving Knowledge. In: Proc. of the International Conference on Building and Sharing Very Large-Scale Knowledge Bases (1995)
18. Naganuma, T., Kurakake, S.: Task Knowledge-based Retrieval for Service Relevant to Mobile User's Activity. In: Gil, Y., Motta, E., Benjamins, V.R., Musen, M.A. (eds.) ISWC 2005. LNCS, vol. 3729, pp. 959–973. Springer, Heidelberg (2005)
19. Naganuma, T., Luther, M., Wagner, M., Tomioka, A., Fujii, K., Fukazawa, Y., Kurakake, S.: Task-Oriented Mobile Service Recommendation Enhanced by a Situational Reasoning Engine. In: Mizoguchi, R., Shi, Z.-Z., Giunchiglia, F. (eds.) ASWC 2006. LNCS, vol. 4185, pp. 768–774. Springer, Heidelberg (2006)
20. Song, Z., Labrou, Y., Masuoka, R.: Dynamic Service Discovery and Management in Task Computing. In: Proc. of the First Annual International Conference on Mobile and Ubiquitous Systems: Networking and Services (2004)
21. Theoharis, Y., Christophides, V., Karvounarakis, G.: Benchmarking database representations of RDF/S stores. In: Gil, Y., Motta, E., Benjamins, V.R., Musen, M.A. (eds.) ISWC 2005. LNCS, vol. 3729, pp. 685–701. Springer, Heidelberg (2005)
22. Van Welie, M.: Task-Based User Interface Design. PhD thesis, Vrije Universiteit, Amsterdam (2001)
23. von Hunolstein, S., Zipf, A.: Towards Task-oriented Map-based Mobile Guides. In: Proc. of the International Workshop HCI in Mobile Guides, Mobile HCI (2003)
24. Weiss, C., Bernstein, A., Boccuzzo, S.: i-MoCo: Mobile conference guide - Storing and Querying Huge Amounts of Semantic Web Data on the iPhone/iPod touch. In: Billion Triples Challenge at the International Semantic Web Conference, ISWC (2008)
25. Chandrasekaran, B., Josephson, J.R.: The Ontology of Tasks and Methods. In: Proc. of the AAAI Conference on Artificial Intelligence (1997)
26. Rajpathak, D., Motta, E., Roy, R.: A Generic Task Ontology for Scheduling Applications. In: Proc. of the International Conference on Artificial Intelligence, IC-AI (2001)
27. Gómez-P'erez, A., Benjamins, V.R.: Applications of Ontologies and Problem-solving Methods. AI Magazine 20(1), 119–122 (1999)

Cinematic Street: Automatic Street View Walk-through System Using Characteristics of Modified Maps

Kaori Kobayashi, Daisuke Kitayama, and Kazutoshi Sumiya

University of Hyogo, Japan
{nd10r013@stshse,dkitayama@shse,sumiya@shse}.u-hyogo.ac.jp

Abstract. Users often use modified maps on Web pages for acquiring information such as sightseeing information and places to go for the first time. However, people cannot visualize the appearance of the station exit and the surrounding buildings. Therefore, we propose a system for transforming modified maps into streaming video based on the features of modified maps. Our system performs analysis of the modified maps and transforms them into streaming video. In this analysis, the system determines the type of modified maps by extracting semantic relations and positional relations among geographical objects on them. In this transformation, the system shows the streaming video by letting film syntax apply to street view based on the type of modified maps. Additionally, we experimented with analyzing the modified maps.

Keywords: modified map, street view, geographical information system.

1 Introduction

Users often use modified maps for personal and business travel because the maps are simplified to suit a specific purpose. People search for these modified maps on Web pages. For instance, when people plan to travel, they might search for tourism-related information on the Web and find modified maps showing sightseeing spots. Additionally, when people go to a university for the first time, they might search the university's Web page and view the map to find directions. In this way, modified maps are used in various situations.

We define modified maps as maps that express geographical objects for a certain purpose. In addition, shapes of buildings and streets on modified maps are more deformed and simplified than on general maps such as Google Maps and Bing Maps. Besides, modified maps have two features: expressing routes and expressing geographical objects. It is possible to classify the modified maps into three types by assortment of the features.

However, we consider that users do not fully understand the geographical information, only the modified map, because the modified maps do not include geographical information from the general maps. Moreover, users may not be able to visualize the appearance of the station exit and surrounding buildings. Therefore, users have to confirm street view and search for photographs to acquaint themselves

K. Tanaka, P. Fröhlich, and K.-S. Kim (Eds.): W2GIS 2011, LNCS 6574, pp. 142–158, 2011.

with local appearances. Consequently, we propose a system for transforming the modified maps into streaming video using street view. We call this system Cinematic Street. We consider that streaming video can provide more geographical information and visual information to users than the modified map. Therefore, users will find it easier to understand information on the modified map. We explain our scenario as follows: Users enter geographical objects' names as query keywords on a Web search engine. The user accesses the Web sites, including the modified maps. Our system analyzes the modified map and transforms it into streaming video using street view. In this analysis, the system determines method for showing the geographical objects of the keywords by analyzing semantic relations and positional relations of the modified map. The procedure adopted to implement this method is as follows.

- analyzing the modified maps based on contained geographical objects
- transforming the modified maps into streaming video using street view

As in related works about modified maps, users are able to easily create the modified maps using Destination Maps of Bing Maps [1]. The generation of modified maps such as this service has been studied extensively [11] [18]. Fujii et al. [10] proposed a method for the generation of guide maps for mobile devices. They analyzed the features of the guide maps and classified the maps as route-type, survey-type, and deformation-type maps. They did not analyze the type of modified maps in any of these cases, because they focused only on the generation of modified maps. However, it is necessary to analyze the type of modified maps to show the streaming video based on the type of modified maps.

Street view is the one of the available tools for expressing information about modified maps, because it can display the local geographical information [1] [4] [5] [6] [13] [16] [17]. Street view is often used to explore locations by walking and describing real space in a virtual space. Further, street view has some other attractive services [7] [15]. In this way, street view is an attractive new medium. However, street view is not effectively utilized by users. We utilize street view to transform the modified map into the streaming video. However, in either case, modified maps are only generated and street view remains the content of only street view and general maps. We try to think ahead about the future of the modified maps and street view.

In Section 2, we provide an overview of our approach. In Section 3, we discuss the extraction relations among the geographical objects on the modified maps and the determination of types of the modified maps. In Section 4, we describe ways to employ street view based on the types of modified maps. In Section 5, we discuss our experiments. Finally, in Section 6, we conclude the paper and discuss future works.

2 Our Approach

2.1 Definition of Geographical Objects

Here, we define geographical objects as the place names on modified maps, with the following attributes: name, position on the map, multiple categories, and a type (path or node). For example, Market Street is the geographical object in San Francisco and is included in categories such as "Streets in San Francisco, California," "Shopping districts and streets in the United States," and "Lincoln Highway." On the other hand, the University of California, Berkeley, is included in categories such as "Educational institutions established in 1868," "National Register of Historic Places in the San Francisco Bay Area," and "California Historical Landmarks." Our system identifies roads, streets, and roadways as paths when geographical objects belong to the category including keyword "Street." Targeted geographical objects such as stations, temples and shrines, universities, and institutions, are called nodes. Nodes are geographical objects, which exclude paths. In the abovementioned examples, the former categories refer to the categories of paths and the latter are examples of nodes.

These categories are conceptual hierarchies with a DAG (directed acyclic graph) structure. We utilize these conceptual hierarchies to extract equivalent relations among geographical objects. Concepts with the same hierarchical level have the same level of abstraction. In this work, prefectures and municipalities, mountains, and rivers are excepted from geographical objects.

2.2 Outline of Cinematic Street's Procedures

Users access Web pages that include modified maps by entering keywords that are geographical objects name on a Web search engine. In this regard, our system treats only the geographical objects' name in any keywords. After that, users select the modified maps that they want to view as video stream. We explain overall procedure of Cinematic Street in Figure 1.

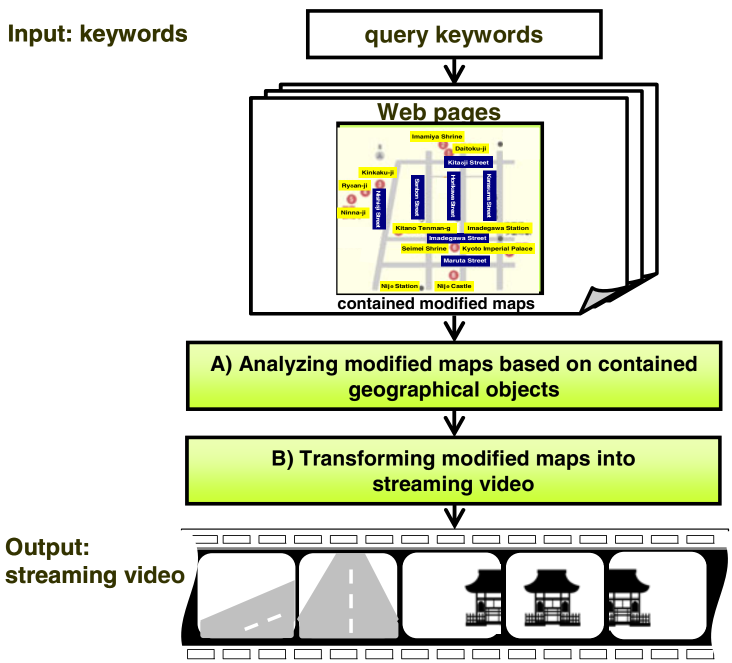


Fig. 1. Outline of Cinematic Street

A) **Analyzing modified maps:** The system extracts the geographical objects' name from the modified map. It extracts semantic relations among the geographical objects and determines the types of maps using positioning relations among them.

B) **Transforming modified maps into streaming video:** The system determines the styles of showing the streaming video based on the types of modified maps. It acquires partly necessary street view points in each type and generates the streaming video by connecting each street view. Additionally, it expresses the relations among geographical objects by using film syntax. It partly generates the streaming video based only on the keywords entered by users.

3 Analysis of Modified Maps

In this section, we discuss the definitions of the equivalent relations among node objects and the extraction of these relations using conceptual hierarchies of node objects. In addition, we present the determination of type of modified maps using positional relationships among geographical objects.

3.1 Concept of Analysis of Modified Maps

We consider that modified maps have the following two features: the feature of expressing node objects and the feature of expressing routes. By the former feature, the node objects are expressed mainly on the modified maps. Thus, several equivalent relations of node objects exist on the modified map. On the other hand, by the latter feature, routes are expressed mainly on the modified map. Thus, several node objects exist around path objects on the modified map. Therefore, it is necessary to extract the equivalent relations of the node objects. Our system determines the map types by analysis of the ratio of the equivalent node objects and the ratio of the general node objects and paths to confirm the existence or nonexistence of the map features.

3.2 Extraction of Equivalent Relations among Nodes

Here, we provide the definition of equivalent relations. Equivalent relations among nodes imply that the concepts of the nodes are semantically the same. Thus, when there exist nodes at the same hierarchical level that represent the perspective intended to be shown by a maker, the relations among these nodes are equivalent relations. Therefore, relations among nodes differ for each modified map. We can regard the theme of the class of the modified map category as a typical node class on the modified map.

We describe the process of extracting the relations among nodes. We consider that even if the same nodes appear on several modified maps, these objects do not always have equivalent relations. For example, in the case of Figure 2, "Temple A" and "University B" appear on two modified maps. Here, "University B" is an old historic piece of architecture. In these maps, equivalent relations of two nodes differ by the types of surrounding nodes. In the left modified map, we can assume that the two nodes have equivalent relations because the map shows the positional relations of the nodes involving "cultural heritage." On the other hand, in the right modified map, we can assume that the two nodes do not have equivalent relations because the map shows positional relations of nodes of "educational institutions."

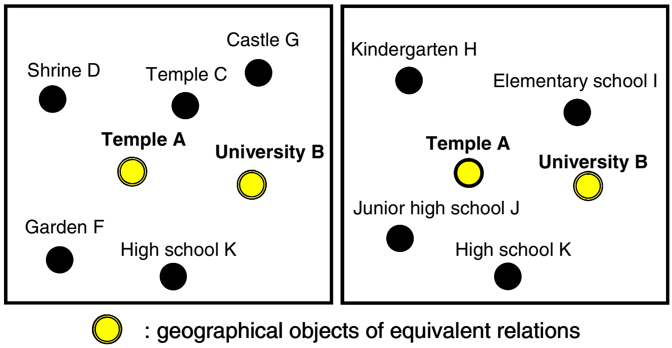


Fig. 2. Examples showing that equivalent relations among nodes are different for different modified maps

Consequently, we first extract the typical categories of the modified map. Next, we extract the nodes at the same distance from the typical categories as equivalent relations. Additionally, there are other relations among nodes, such as landmark relations and inclusive relations; however, we focus only on the equivalent relations in this paper.

The specific procedure for extracting the relations among nodes is as follows:

I. The categories to which nodes on a modified map belong are determined, as shown in Figure 3.

II. The weights are added from all descendent nodes ($o_i \in O$) to each category ($a_j \in A$). The weights of the category (*Category* (a_j)) are determined by each distance from descendent nodes, which are elements of the category ($dis(a_j, o_i)$). When a category is close to a node, its weight is high. On the other hand, when a category is distant from a node, its weight is low. The category value is calculated by adding all the weights corresponding to the category (Figure 3).

$$Category(a_j) = \sum_{o_i \in O} \frac{1}{dis(a_j, o_i)} \quad (o_i \text{ is } a\text{'s descendant}) \tag{1}$$

III. When a value is higher than the threshold, the system determines the typical categories. In Figure 3, a typical category is "National treasures."

IV. When there are nodes at the same hierarchical level from a typical category, they have equivalent relations in the typical category. In Figure 3, nodes having equivalent relations are Castle G, Temple A, Shrine D, and University B.

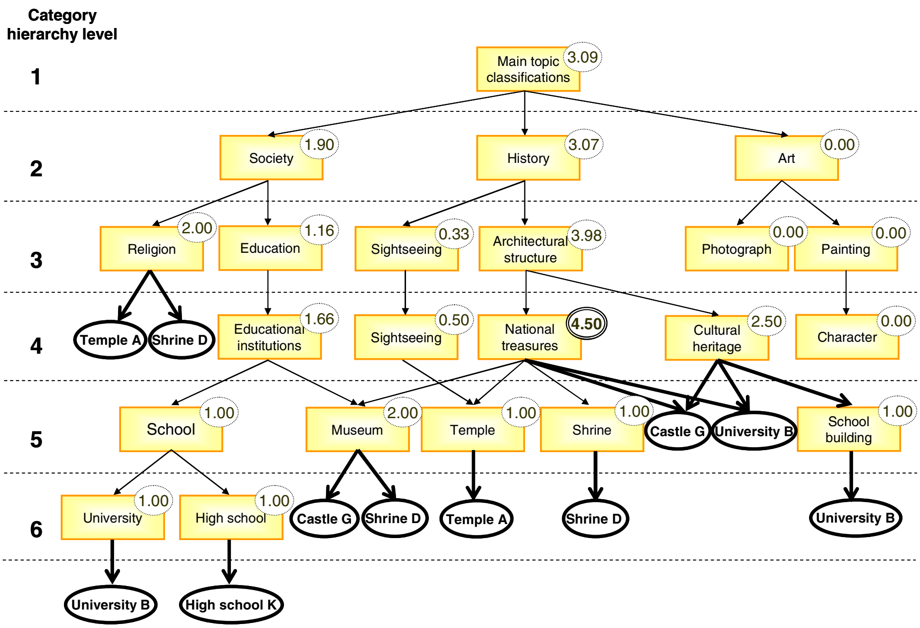


Fig. 3. Calculation of category values as typical categories on the map

3.3 Determination of Types of Modified Maps

We describe the approach to determine the type of modified maps. We consider that we can determine the features of modified maps by classifying the sets into paths and nodes; this is because we believe that the positioning of paths and nodes differs according to the features of the maps. For instance, if many nodes that are not bound by the equivalent relations are shown and connected by many paths, the modified map shows routes among the nodes. On the other hand, if few paths are shown and many nodes with equivalent relations are shown, the modified map shows the positional relations among specific nodes.

3.3.1 Procedures for Determining Modified Map Types

We first present the features of each modified map. Route maps are modified maps that show some paths between two nodes; hence, paths are important for route maps. Object maps are modified maps that show various nodes; hence, nodes are important for object maps. Mixed-feature maps are modified maps that show nodes existing in the vicinity of paths; hence, both paths and nodes are important for mixed-feature maps.

The system creates sets of geographical objects in the modified map to detect their features. Accordingly, we consider that detection of the type of modified maps is possible in keeping with the path and node trend of each set. We show the procedure below.

I. The minimum spanning tree is composed of the layout of the paths and nodes, as shown in Figure 4(a).

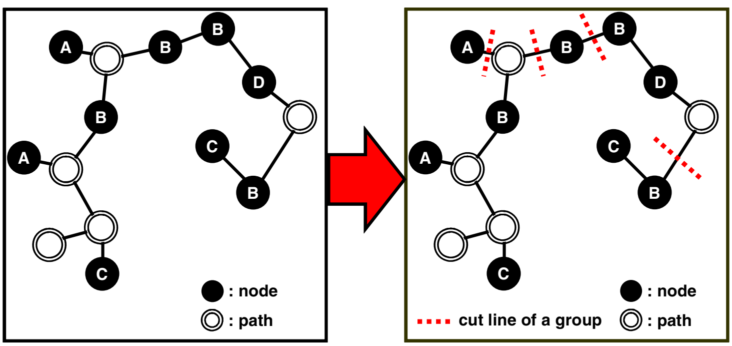
II. We assume the minimum spanning tree as an order tree. Determination of ordering is a preorder traversal based on the order tree. Geographical objects range in an increasing numerical order from left to right in the tree.

III. The first *node* or *path* is added to Set_j. If there are other elements under the elements of Set_j, the system considers the next geographical object.

IV. If the next geographical object is $path_{i+1}$, it is added to Set_j. If the next geographical object is $node_{i+1}$ and the relations among the elements of Set_j and the *node* are equivalent, $node_{i+1}$ is added to Set_{j+1}. If there are other elements under Set_{j+1}, the system considers the next geographical object.

V. If there are no other elements under the elements of *Set*, the *Set* is closed.

VI. Steps 4 and 5 are repeated until all geographical objects are covered.



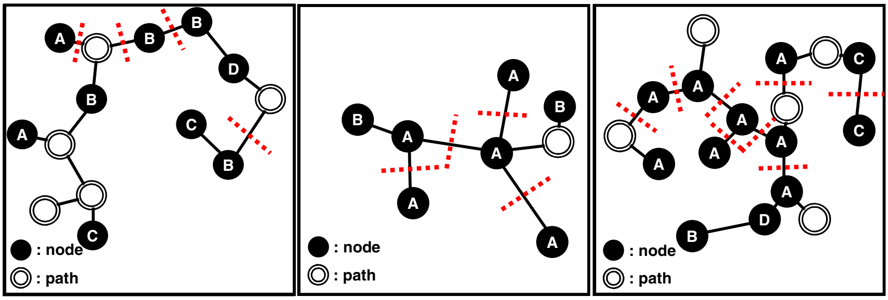
(a) Minimum spanning tree (b) Determination of creation of sets

Fig. 4. Creation of sets of the minimum spanning tree

Figure 4(b) illustrates the creation of sets from node C on the left. The letters denote the equivalent relations among nodes. On the basis of sets, our system detects modified maps as a “route map,” “object map,” and “mixed-feature map.”

3.3.2 Determination of Each Modified Map Type

Route map: The important factors in the route map are paths. Therefore, the system calculates the number of all geographical objects in the modified map and the number of geographical objects in the sets that include paths. When the number of geographical objects in the sets is above a threshold, the map is considered as a route map. We decided the threshold value to be half of the number of all geographical objects determined by the preliminary experiments. In the case of Figure 5(a), the total number of geographical objects is 14, so the threshold value is 7. The number of geographical objects in 3 sets that include paths is 12 and is larger than the threshold. Therefore, this modified map is considered as a route map.



(a) route map (b) object map (c) mixed-feature map

Fig. 5. Example of determination of type of modified maps

Object map: The important factors in the object map are nodes. When the number of sets is above a threshold of the number of geographical objects, the map is determined as an object map. In Figure 5(b), the first node is node B on the left. The number of geographical objects is 8, and the number of sets is 5. The map is not a route map, because the number of geographical objects that include paths is 1. Moreover, the number of sets fulfill the threshold. Therefore, the modified map is considered as an object map.

Mixed-feature map: Both paths and nodes are important factors in this map. When the map meets the requirements of being both a route map and an object map, it is considered as a mixed-feature map. In Figure 5(c), the first node is node A on the left. The map satisfies the requirement of being an object map because the number of geographical objects is 17 and the number of sets is 9. Therefore, the map is an object map. In addition, the map meets the requirement of being a route map because the number of geographical objects in sets that include paths is 13. Accordingly, the map is considered as a mixed-feature map. In addition, if none of the above apply, the system considers the type of map as being "others."

4 Transforming Modified Maps into Streaming Video

In this section, we discuss transformation of the modified map into streaming video based on the map type. The conventional styles of representing a video based on the film syntax are applied to street view. Firstly, we discuss adaptation to street view of the film syntax. Secondly, we describe styles of showing streaming video based on the types of the modified map. Lastly, we explain an application of the significant impact of the relations among geographical objects.

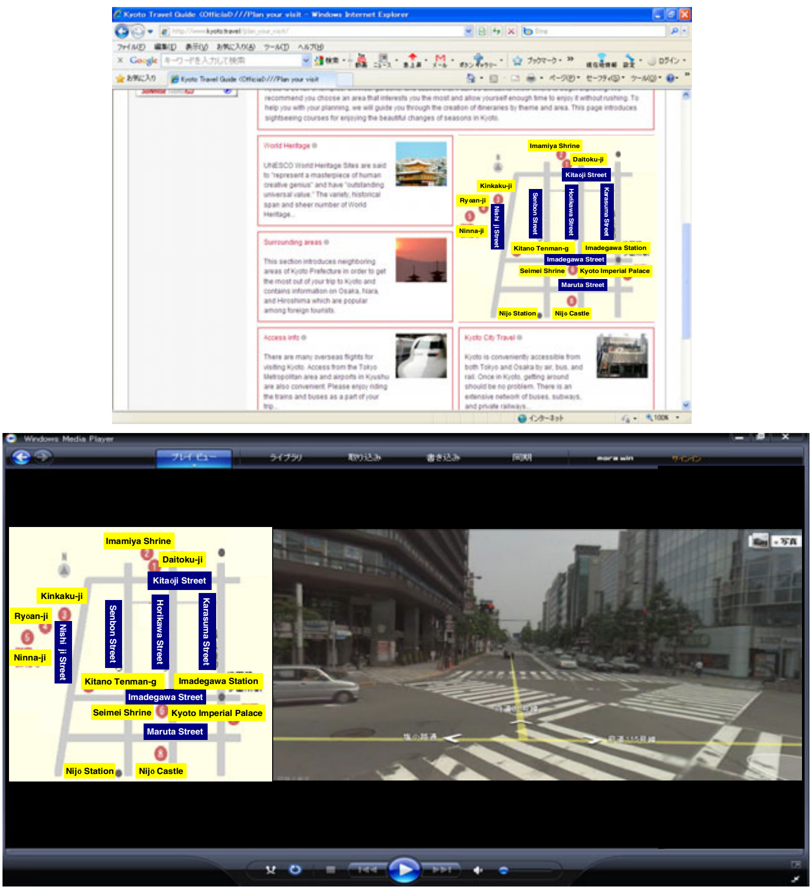
4.1 Application of Film Syntax to Street View

It is necessary to express the relations among geographical objects because paths and nodes are different factors, and there are equivalent nodes and general nodes. Accordingly, we describe the styles for representation among geographical objects.

There are various methods for representing connecting shots in a film [20] [21]. In our research, we treat shots as movements of street view. We utilize three transitions for connecting street view and four shots for representing streaming video using street view. Three transitions are “fade-in,” “dissolve,” and “wipe.” Four shots are “180° line,” “establishing shot,” “pan,” and “graphic match.”

Paths and nodes clearly show differences in the factors by utilizing “fade-in” in each shot line because we want to depict the passage of time and shift in location. For nodes with equivalent relations, the system shows that a change is small by utilizing the “dissolve” effect at each turning point; this is because we want to depict a smooth change of the scene. For nodes without equivalent relations, the system quickly shows a change using the “wipe” effect; this is because no relations are to be shown among the nodes.

A Web page which contains modified maps



Transformed streaming video

Fig. 6. Screen image of interface of Cinematic Street

With regard to the display range of the streaming video, in the case of the street view of paths, pictures of the streaming video are locked in the traveling direction and show a pathway based on the "180° line." In the case of the street view of a departure place, the picture of the streaming video is gently turned by 45° to the right and left in front of a departure place using the "establishing shot." If node objects of equivalent relations and same sets continue, the system expresses that their relation and set are the same by using the "graphic match." In the case of the street view of a node, the system shows the node of a certain uniformity range using "pan." A certain uniformity range is a range on the road that corresponds to an online map with many photographs, and the system focuses on the traveling direction in this range. In addition, when the street view is located in the range of a road where numerous photographs exist, the system turns from the progress direction to the node direction and displays this turn.

The targets for representation in the film are "subjects," "characters," and "objects." On the other hand, targets in our work are "paths" and "nodes." We apply styles of film to street view to represent the features of maps. The following are the types of street view effects obtained in each modified map using our system. Figure 6 shows the interface of Cinematic Street.

4.2 Route Map

A streaming video of a route map shows the route guide because the map has the feature of expressing routes. If the user inputted the geographical objects' names as one query keyword, our system shows a partial route guide video to go to the spots of geographical objects. If the user inputted the geographical objects' names as more than one query keyword, the system shows a partial route guide among their keywords. Anyway, it is important to show the routes between a departure node and an arrival node. It is necessary for users to comprehend order and image of a departure node, streets, and an arrival node. Therefore, the system shows a sufficient street view surrounding the departure node, corners, and the arrival node, and shows the street view of the traveling direction in the streets.

We explain the procedure for generating the video stream from a departure node to another node when a user selects a departure node.

I. When the order of geographical objects is "node, path, node," the system searches for a route. Next, if nodes exist after the path, then multiple nodes are allowed.

II. The system adds the shot to the streaming video that street view is moved across the surroundings of the departure node from left to right as in the "establishing shot."

III. The streaming video of the paths is aimed toward the corner or the arrival node and advances in steps of three toward the destination; it uses effects such as "180° line" and "tracking shot."

IV. The streaming video shown in the third step is moved from the corner or the arrival node and advanced toward the corner or the arrival node with effects such as the "180° line" and "tracking shot."

V. If the next point is a corner, the system moves to step VI, and if next point is the arrival node, the system moves to step VII.

VI. The street view of the corner is shown from left to right using an effect such as "pan."

VII. The street view of places close to the arrival node is shown using effects such as the "establishing shot."

VIII. When this process is completed, the system moves to the next departure node and follows the same steps.

4.3 Object Map

A streaming video of an object map shows nodes' introduction because the map has the feature of expressing geographical objects. If a user inputted the geographical objects' names as one query keyword, our system shows in random order nodes of equivalent relations with the keyword node. If a user inputted the geographical objects' names as more than one query keyword, the system shows, in order of decreasing strong relations, nodes of equivalent relations with the keywords' nodes. Therefore, the system shows enough street view surrounding each node. An object map is needed to show a relationship among nodes because nodes are important factors in an object map. If nodes of equivalent relations continue, the system shows appearance of nodes using the same angle of street view.

When a user selects a departure node, we explain procedures generating the video stream from a departure node to another node.

I. The streaming video shows the departure node and equivalent relations nodes in order by using an "establish shot."

II. The nodes of equivalent relations can be justified by Cinematic Street using a "graphic match."

III. The streaming video is used to show the appearance of nodes using the "pan."

IV. Nodes that do not have equivalent relations at random are shown.

4.4 Mixed-Feature Map

A streaming video of a mixed-feature map shows nodes' introduction of the surrounding path because the map has the features of both. If a user inputted the geographical objects' names as one query keyword, our system shows a set including the keyword node based on paths of the set. If a user inputted the geographical objects' names as more than one query keyword, the system shows the sets including each node object as the user inputted query keywords based on paths of each sets. It is important to know which path sets the nodes are included in. If streaming video is generated in only one set, all geographical objects are shown as street view of the same angle to show that these are same sets. However, if streaming video is not only in one set, in the change between sets, the angle between geographical objects differs.

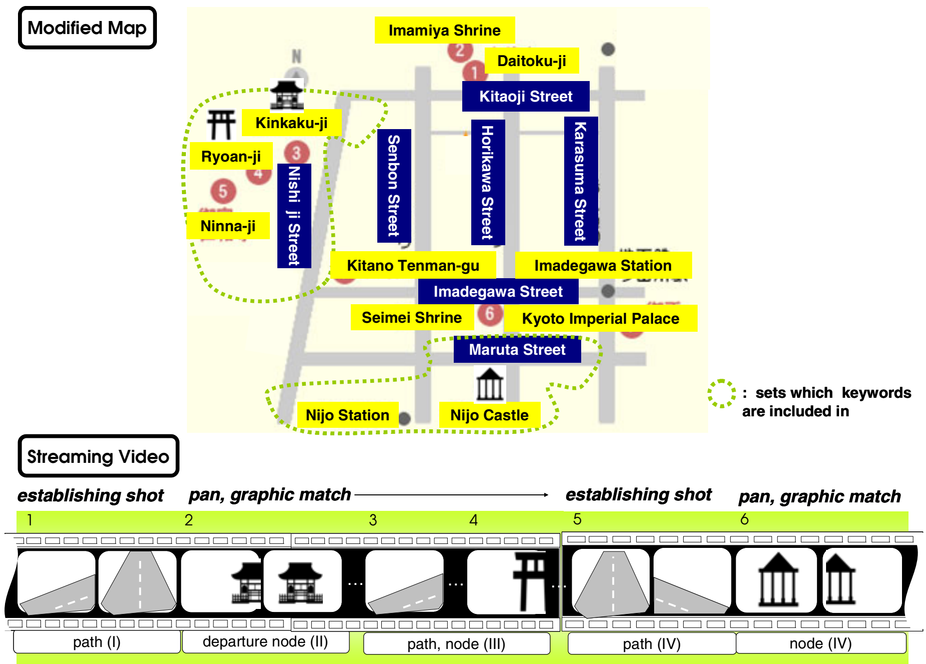


Fig. 7. Example of outputs captured by Cinematic Street

In the case of a modified map determined to be a mixed-feature map, the system forms several sets based on paths. The system shows a streaming video of sets of keywords entered by a user. This figure of transformed streaming video is a portion of expressing streaming video. The top characters are shots of street view in the case of each geographical object. The arrow indicates the continuation of the same shot. Numbers represent the order in which geographical objects appeared in the streaming video. Illustrations express moving of shots. The bottom characters indicate the type of geographical objects and positions in the step of mixed-feature maps.

When a user selects a departure node, we explain the procedure for generating the video stream from a departure node to another node.

I. The path of a set including a departure node is shown using an "establishing shot," as shown in No. 1 in Figure 7.

II. The node of a same set is shown using "pan" and "graphic match," as shown in No. 2 in Figure 7.

III. Steps I and II are repeated until the last node in a set is over, as shown in Nos. 3–4 in Figure 7.

IV. When the first set is finished, the next set is shown, as shown in Nos.5–6 in Figure 7.

5 Evaluation

We performed an evaluation experiment for determination of modified map type. Additionally, we confirmed effectiveness of using semantic relations and positional relations for determination of map types. The participants were seven university students, and they looked 16 painted modified maps. We requested the participants to answer the following two questions about each modified map: Question 1: Do you think this modified map shows routes among nodes? Question 2: Do you think this modified map shows similar nodes? Thus, Question 1 is a question about the feature of expressing routes, and Question 2 is that about the feature of expressing geographical objects. The answer is supposed to be either "yes" or "no." We show the experimental results in Tables 1 and 2.

The left column is the number of modified maps. Columns of Participant's answer are percentages of answers "yes". The right column is the result of determined map types by our system. We use participants' answers as correct answer that has ration more than threshold (0.50). Bottom line is accuracy which is ratio of correct answers in result of the system. From a result, we found that accuracy of our system is reasonably well, because accuracies of two features are 70% each them.

As examples of good results particularly, there are map 2, 5, 10 and 14. The result of map 2 matched the result of system in the feature of expressing routs, because there were not similar buildings in the map, and paths names were written down clearly. The result of map 5 matched the result of system in the feature of expressing geographical objects, because paths were very rough and not detail, and many temples were written down. The result of map 10 and 14 matched the result of system in the both features, because paths names were written down clearly and many universities were written down.

Table 1. Experimental result of determination of two features

	u			
		s s		s s
1	7		1	
	71			
4				
	71		1	
7	4			
			4	
	7			
1				
11	71		4	
1			7	
1				
14				
1	7		4	
1				
u	7		7	

As examples of bad results particularly, there are map 4, 6, 8, 11 and 16. The system determined that the map 4 has the feature of expressing routes, but participant did not think so. The reason is thought that the participants can recognize detailed path or not detailed path, but the system can not recognize them. The map 16 also is same reason. The system determined that the map 6 and 8 has the feature of expressing geographical objects, but participant did not think so. The participants answered that the map 6 has the both features. The reason is thought that the participants can recognize paths of figure, but the system can not recognize them. Additionally, the participants answered the feature of expressing route more than the feature of expressing geographical objects. The reason is thought that the participants do not view stations as buildings, but the system determines stations as buildings. The map 11 also is same reason. Therefore, value of object is low.

In Table 2, participant's map type columns are percentages of determined map type by the result of a question. In this case, if the responses to Questions 1 and 2 are yes and no, respectively, the map is a route map. On the other hand, if the responses to Questions 1 and 2 are no and yes, respectively, the map is an object map. If responses to both questions are yes, the map is a mixed-feature map, and if responses to both questions are no, the map is "others." In Table 2, we use a map type as correct answer that has the maximum ratio.

Table 2. Experimental result of determining the types of modified maps

	c ec				em
	e	ec	m xe	e	
ma	0.00	0.	0.5	0.00	e
ma	0.8	0.00	0.00	0.	e
ma	0.	0.00	0. 9	0. 9	e
ma	0. 9	0.00	0.00	0.	e
ma 5	0.00	0.5	0. 9	0.	e
ma	0.00	0. 9	0.	0.00	e
ma	0.00	0.	0.	0.	e
ma 8	0.	0.00	0.	0.	e
ma 9	0.	0.	0.	0.00	m xe
ma 0	0.	0.	0.	0.00	m xe
ma	0.	0.	0. 9	0.	m xe
ma	0. 9	0.00	0.5	0.	m xe
ma	0. 9	0.00	0.00	0.	e
ma	0.00	0.5	0. 9	0.	ec
ma 5	0. 9	0.	0. 9	0. 9	ec
ma	0.00	0.00	0.00	.00	e
acc ac	0.	0.50	0. 5		

We compared the result of correct answers with the result of our system. In the map 1-4, 13 and 16, the system determined that each map is a route map, and route map in the result of participants is slightly higher than the other three map types, but there are few differences among them. As a result, only map 2 was consistent with the results of participants. In the case of map 1, the problem is threshold of determining map types. In the case of map 4, participants thought that paths are too rough on the maps, but the system cannot determine that paths on the maps are as detailed as routes. In the map 5-8, 14 and 15, maps that were consistent with results of participants were few. The main reason the system cannot recognize paths of figures is because if there are no path names, participants trust paths of figures. Moreover, the system determines stations as buildings, but participants answered that there are no similar buildings because they do not recognize that stations are buildings. The system could not determine that the maps are “others.” However, a common point of the maps in which participants judged to be “others” is that paths are too rough. Moreover, we found that participants recognized the rail track as routes in map 15, and the result of map types included all types because an understanding of similar nodes was different among the participants. The results of map 9-12 are not too bad. However, particularly, in the map 11, the participants do not feel that stations are similar buildings, because the participants stations as another factors. The system determined that stations are similar buildings. Additionally, the results of map 3, 7, 8, 11, 12 and 15 are various. A common point of these maps is included many stations. It is necessary to determine stations as another factor.

As these result, in route map, the maps that paths of figures and paths name of characters are written down are determined as route map. On the other hand, in object map, the maps that paths is rough and not detail, and there are uniform buildings such as universities and temples are determined as object map. Mixed-feature map needs above conditions. In addition, object map and mixed-feature map would be better off the map that stations are included.

In this experiment, we thought that classification of various modified maps is almost covered by our determination of map types. We focused that there are nodes of no equivalent relations along the paths on modified maps in route map. Therefore, even if the system can not recognize paths of figure, the system can determine route maps. Additionally, we think that object map is included many figures and icons. However, we focused that there are many nodes of equivalent relations on modified maps in object map. Consequently, even if the system can not recognize figures and icons, the system can determine object map. Thus, in mixed-feature map, the system can also determine the map. Of course, if the system can recognize figures and icons, accuracy for determining type of modified map in the system will get even better. However, we found that the system can enough determine types of modified maps using only semantic relations and positional relations. Therefore, we are going to consider about other relations among geographical objects and other features of modified maps.

6 Conclusions

We proposed a concept of a system for transforming a modified map into a streaming video based on the features of the modified map. In this paper, we described the extraction of equivalent relations among geographical objects in order to determine the type of each modified map on the basis of its features. Moreover, we described styles for representing a streaming video based on the type of the modified map. Additionally, we discussed experiments for determining types of modified maps.

There exist automatic methods for presenting visual information from various multimedia contents. For instance, there are methods for presenting a Web page based on relations among Web pages [8] and methods for presenting photographs based on semantic relations [12]. In these fields, camera movement is not required, because the images are still images. However, street view requires camera movement because it shows a direction. On a related video picture, there are various film languages [20] [21]. Additionally, there are several methods for showing video [3] [9]. When interactive tasks are performed, some systems concentrate on finding the best camera placement [2] [19]. He et al. [14] proposed a Virtual Cinematographer in which the virtual architecture is suitable for some real-time applications and is well related to the virtual actors. In these researches, target objects are moving objects such as characters and a camera can recognize target objects. However, target objects in street view in this research are geographical objects that do not move. In addition, street view cannot recognize target objects. Therefore, it is necessary to develop styles for recognizing target objects in street view. We utilize film syntax as automatic styles for presenting streaming video using street view.

In the future, we intend to develop the system and examine the effectiveness of the styles for showing street view based on the type of each modified map. In addition, we are going to discuss other relations among geographical objects, such as those among landmarks and inclusive relations.

Acknowledgments

This research was supported in part by a Grant-in-Aid for Scientific Research (B)(2) 20300039 and a Grant-in-Aid for JSPS Fellows 21.197 from the Ministry of Education, Culture, Sports, Science and Technology, Japan.

References

1. Bing Maps Beta, `http://www.bing.com/maps/`
2. Phillips, C.B., Badler, N.I., Granieri, J.: Automatic viewing control for 3D direct manipulation. In: Zeltzer, D. (ed.) Computer Graphics (1992 Symposium on Interactive 3D Graphics), vol. 25, pp. 71–74 (1992)
3. Christianson, D.B., Anderson, S.E., He, L.-w., Salesin, D.H., Weld, D.S., Cohen, M.F.: Declarative camera control for automatic cinematography. In: Proceedings of the AAAI 1996 (1996)
4. EveryScape.com, `http://www.everyscape.com/`
5. Google Maps Street View, `http://maps.google.co.jp/`
6. goo lab. Walkthrough System, `http://map.labs.goo.ne.jp/walkthrough/index.php`
7. historypin, `http://www.historypin.com/`
8. Sumiya, K., Takahashi, M., Tanaka, K.: Summarization and Presentation for Web Documents using Context Paths. In: Proceeding of the IASTED International Conference on Information Systems and Database (ISDB 2002), pp. 169–177 (2002)
9. Kennedey, K., Mercer, R.E.: Planning animation cinematography and shot structure to communicate theme and mood. In: International Symposium on Smart Graphics (2002)

10. Fujii, K., Sugiyama, K.: Route Guide Map Generation System for Mobile Communication. Transactions of Information Processing Society of Japan 41(9), 2394–2403 (2000)
11. Maruyama, K., Tanizaki, M., Shimada, S.: Road Network Normalized Shaping Model for Generating Deformed Maps and Its System Evaluation. Transactions of Information Processing Society of Japan J87-A(1), 108–109 (2004)
12. Yatsugi, K., Fujimura, N., Ushiama, T.: A Web-Based Approach for Automatic Composition of an Insightful Slideshow for Personal Photographs. In: Velásquez, J.D., Ríos, S.A., Howlett, R.J., Jain, L.C. (eds.) KES 2009. LNCS, vol. 5712, pp. 623–630. Springer, Heidelberg (2009)
13. Live Search Maps, `http://preview.local.live.com/`
14. He, L.-w., Cohen, M.F., Salesin, D.H.: The virtual cinematographer: A paradigm for automatic real-time camera control and directing. In: Proceedings of the ACM SIGGRAPH 1996, pp. 217–224 (1996)
15. Navitte!, `http://navitte.jp/route387.html`
16. Photosynth, `http://photosynth.net/`
17. PhotoWalker, `http://web.sfc.keio.ac.jp/~htanaka/earthwalker/photowalker/`
18. Nagata, T., Maeda, Y.: Agent-based Geographic Information Selection in the Deformed Map. In: Human Interface Symposium 2006, pp. 479–484 (2006)
19. Bares, W.H., Lester, J.C.: Cinematographic User Models for Automated Realtime Camera Control in Dynamic 3D Environments. In: Proceedings of the Sixth International Conference on User Modelling, pp. 215–226 (1997)
20. Imaizumi, Y.: Film Syntax: Shot Analysis of Japanese Film. Sairyu-Sha (2004)
21. Leo, B., Marshall, C.: Film Theory and Criticism: Introductory Readings, 6th edn. Oxford University Press, New York (2004)

HYDROSYS – A Mixed Reality Platform for On-Site Visualization of Environmental Data

Antti Nurminen, Ernst Kruijff, and Eduardo Veas

Helsinki Institute for Information Technology
antti.nurminen@hiit.fi,
http://www.hiit.fi
Graz University of Technology
kruijff@icg.tugraz.at,veas@icg.tugraz.at,
http://www.tugraz.at

Abstract. Modern mobile devices and networks facilitate the development of increasingly graphical and complex mobile GIS or *geomedia* applications. We advance the state-of-the-art with the HYDROSYS system, a heterogeneous platform integrating multiple data sources and components, allowing interactive environmental data analysis and monitoring in the field. Our key contributions are 1) scalable on-the-fly streaming of environmental sensor data, 2) highly graphical mixed reality representations embedding multivariate sensor data visualizations, including access to results of near real time simulations and other physically-based computations, 3) support for semantically meaningful public participation. We balance the use of standardized formats and optimized mobile transmissions. Our system builds on use cases ranging from hydrological professionals and municipal authorities to the environmentally-aware public. We present the challenges, solutions and implementation of the system, providing observations on potential bottlenecks.

Keywords: Mixed / Augmented Reality, 3D maps, sensor networks, mobile applications.

1 Introduction

The natural environment is undergoing dramatic changes. As environmental awareness increases, accurate and up-to-date information from the state of the environment is becoming a necessity to support mitigation of problems in an appropriate way. However, environmental monitoring processes are still mostly based on legacy measurement systems. In practice, only weather stations and localized forecasts have reached the level of near real time updates, where results are immediately available to the public. Other environmental data, such as water quality, is still either measured manually, or logged at automated stations into local memory, to be read only between relatively long intervals. However, it has been shown that many significant phenomena may be near instantaneous. If a land mass falls into a river, or a poisonous substance is leaked

K. Tanaka, P. Fröhlich, and K.-S. Kim (Eds.): W2GIS 2011, LNCS 6574, pp. 159–175, 2011.

to the water-ecosystem, only a continuous and frequent measurement can reveal the event. Moreover, if immediate action is needed, such as a flood warning, this information needs to be delivered instantly.

While environmental monitoring can be performed in part in office, this is mainly a passive observation. In active use, the environmentally aware user (public or professional) may pose inquiries on demand *in situ*. Moving in the environment, the user may make observations and require further information. She could make a qualitative annotation of her observation or note down ideas for mitigation, acting as a mobile human sensor. An environmental scientist could even carry mobile sensing devices, providing quantitative measurements: such measurements can possibly be used as input to physical environmental process models. Thus, instant availability of data may provide an answer, or rise further interest to the situation.

Modern mobile technology enables the development of intuitive and highly graphical interfaces. For example, *Google Earth* provides a platform whereby realistic aerial image embedded terrain models are transmitted progressively, and onto which geodata can be overlaid [14]. Having real world content within a virtual environment is called *augmented virtuality* (AV). When one overlays computer-generated graphics onto the real world, enhancing the physical world beyond the user's normal experience, it is called augmented reality (AR) [4]. Together these systems are called *mixed reality* applications [19], and form the crux of our system. Figure 1 presents our target platforms, both AV and AR.

Fig. 1. Two HYDROSYS mixed reality platforms. (left) A cell phone for augmented virtuality. (right) An ultramobile PC equipped with orientation sensors and camera for augmented reality.

Despite recent technological advances, developing a functional system supporting multi-source sensor data streams, simulations and physically based environmental models, public participation and providing highly graphical, interactive mobile mixed reality interfaces poses a grand challenge at every level. The mere complexity of the system may hide the underlying goals and needs. The present paper addresses these issues by taking a user-centric approach, focusing on a selected scenario. We explain the needs and characteristics of the high level system architecture. Thereafter, the main components are discussed, reflecting the identified requirements and challenges.

Main contributions. We provide several innovations for mobile geovirtual systems within the HYDROSYS framework:

– Scalable on-the-fly streaming of environmental sensor data from sensor stations to mobile platforms, focusing on hydrological cases
– Highly graphical mixed reality representations embedding multivariate sensor data visualizations, including access to results of near real time simulations and other physically-based computations
– Support for semantically meaningful public participation
– Integration of the heterogeneous system, balancing the use of standardized formats and optimized mobile transmissions
– User involvement throughout the system design

2 Previous Work

Mobile GIS extends Geographic Information Systems from the office to the field by incorporating technologies such as mobile devices, wireless communication and positioning systems. Mobile GIS enables on-site capturing, storing, manipulating, analysing and displaying of geographical data. Mobile GIS systems include, for example, *FieldWorker, GPSPilot, Fugawi, Todarmal, ESRI ArcPad*, and *MapInfo MapXmobile*. These interfaces and development kits typically faciliatate traditional 2D maps, on top of which multiple types of spatial information can be layered. *FieldWorker* allows online information exchange, and *Todarmal* supports user-defined point, line and polygon based layers.

Geographical interfaces have been moving towards more media-and graphics rich applications for some time now. *Google Earth* can be seen as a generic 3D GIS platform, portraying the environment in 3D, using aerial imagery for landscape detail, with an open format for additional geo-referenced content. It has been extended to mobile platforms with limited features. *Autodesk LandXplorer* is a professional level 3D design tool set for managing large urban environments, supporting standardized file formats such as CityGML, but no mobile version is available.

Mobile navigation systems are also becoming more realistic and (supposedly) intuitive. While common car navigation systems provide a perspective road view, *Navitime* has a rudimentary 3D view and *NaviGenie* provides procedurally textured buildings for urban 3D car navigation. *m-LOMA* is a mobile 3D city map supporting individually textured buildings and dynamic real world entities [22].

Recently, mobile augmented reality has gained interest as an emerging application type. AR systems need accurate *registration* to match graphics with the real world. *Touring Machine* was one of the first mobile AR systems, tracking positioning and orientation with differential GPS and magnetometer [11]. Tracking accuracy can be refined with so-called hybrid-tracking methods [25]. AR systems have been implemented on low-end devices such as cell phones [31] and higher-end platforms, such as *Tinmith* by Piekarski *et. al.* [23] and Schall *et. al.*'s work for civil engineering [26]. To our knowledge, there are no reports of using AR specifically for environmental issues with the exception of ARVINO [16].

3 System Design

Developing future applications is always a dual challenge: how to take a major technological leap while still meeting the needs and expectations of the final application users. Traditional usability research focuses on evaluating user interfaces of the prototypes *after* the system is finished, whereas the developer would need guidelines *before* and *during* research. In our earlier work, we have applied traditional interviews and market analysis, or have left end users completely out of the loop, trusting solely on engineering practices. In HYDROSYS, we attack the dilemma by setting up technological goals but trusting in the *user-centred design method* (UCD) [20,13] for the high-level application design.

Our goal was to build a heterogeneous system with mixed reality visualization aimed at a wide variety of users, including municipal administrators, environmental scientists, construction companies and environmentally aware citizens. To manage this task, we observed and analyzed the common tasks of our end users, interviewed them and in general tried to understand them by studying their cognitive, behavioural, anthropological and attitudinal characteristics. With this empirical foundation, we created *scenarios* and *storyboards* as the medium to communicate between the various stakeholders. As an example, figure 2 presents a simplified, aggregated scenario intended for citizens. In the following developments, we focus on this case.

Fig. 2. A HYDROSYS scenario for environmentally aware citizens

Unna walks in her local environment, enjoying her favourite brook. The river is now dirty, filled with garbage. There is also a truck that has obviously poured something to the river. Unna opens up an mixed reality application, takes a photo of the situation and attaches it directly to the water in camera view, along with a concerned message.

Andy is wondering about the water quality of the small river he came by. His 3D map shows the water murky and bleak. He checks out the water quality graphs showing sudden increase in conductivity. An online hydraulic simulation also tells him the timeline of the event. He clicks on the river for recent annotations and sees Unna's photo from upriver. He comments on her photo, springing up discussion among the local environmental association...

HYDROSYS has run multiple measurement campaigns and presented the system to end users at several stages during development. We have received constructive feedback that has helped us in developing the system further. Similarly, users who are familiar with industrial products, have received a good view on how new, advanced systems are developed. Involving users has prevented the developers to dig too deep into their software. On occasion, they have noted obvious things that developers had somehow overlooked.

3.1 System Requirements

The presented scenario (Fig. 2) hints that our mixed reality applications should provide

- Support for near real time environmental sensor data streaming
- Sufficiently accurate 3D visualization or tracking of device orientation to pinpoint small scale features from the environment
- Visualization of sensor data as part of the environment or as an overlay, thus mixing various representations of sensor data
- Ability to connect to results of simulations or other physically based computational models
- Ability to annotate a semantically interesting entity (such as a brook), not just a point of interest

Implicitly, we can assume the usual requirements on interactivity for the applications. We can also claim recognizability of the environment for a 3D visualization and suitable data structures to support geographically and topologically defined entities and annotations. The field of GIS is in transition from using solely 1D and 2D data visualizations to including 3D visualization. As such, purely numeric table data as well as plots need still be provided as the understandability of data is often connected to user conventions. However, emerging systems tend to suffer from technical or usability limitations, which affect the experience more than convention deviations. Furthermore, given an outdoor environment and the fact that our users simply wandered by and expected the application to be ready, we should also support relatively large areas, requiring out-of-core data management for the mixed reality clients, including networked streaming of map data.

These seem rather conventional requirements for virtual environments or augmented reality. However, our environment is a natural one, without definite boundaries: most current virtual and augmented reality environments are bound to controlled (laboratory) situations. We expect to be able to define connected segments of the environment as whole entities. Furthermore, we need to be able to modify the look of environment directly based on near real time sensor data. And finally, we expect all this to run on mobile devices.

4 System Architecture

Figure 3 presents the main HYDROSYS components. In a heterogeneous system, one of the main issues is the middleware. Many solutions exist, and standardization usually follows research with a delay. We choose the Global Sensor Network (GSN) [2] to stream data from static and possibly dynamic sensor sources.

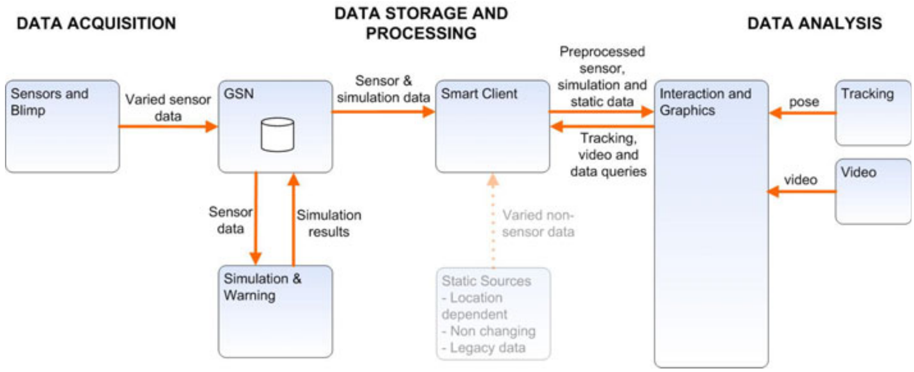


Fig. 3. The HYDROSYS architecture. Data Analysis incorporates the mixed reality clients.

Given the goal of graphically demanding mobile mixed reality applications, we expect all static data to be optimized for rendering prior to transmission or local installation. For dynamic data, such as our sensor streams, simulation results or annotations, we need to consider on-the-fly compression, as we cannot always expect robustness or even mediocre transmission speeds from wireless networks. We opt for an architecture, where static data is hosted and dynamic data may get optimized in *Smart Clients*. A Smart Client acts as a gateway to the mobile devices, delivering data in appropriate format for the mixed reality applications for analysis in the field, accommodating for the limitations of mobile networks (Fig. 3).

4.1 Mixed Reality Clients

HYDROSYS builds its two mixed reality visualizations upon two existing code bases. The cell phone based augmented virtuality (3D map, geovirtual environment) system extends the *m-LOMA* [22] prototype. The handheld computer based augmented reality side, where 3D content is overlaid over video feed, extends *Studierstube* [27]. Both code bases are heavily modified to suit outdoor natural environments with our requirements. 1D and 2D data are directly tied to the mixed reality representations, thus forming a unified multivariate sensor data representation framework.

The challenges in the two types of visualizations are somewhat different. In augmented reality, the main challenge is in registering the device's (or the input camera's) position and orientation accurately, whereas an interactive near realistic 3D view takes the computational resources of mobile devices to their limits.

We now briefly present our two client systems. The cell phone platform has been implemented with C/C++, including the Smart Client and the mixed reality application. We use OpenGL ES 1.0 for 3D rendering, and are currently porting the client to OpenGL ES 2.0. We support Nokia Symbian S60

(N93, N95) and Maemo (N900) smartphones, and desktops (Linux, Windows). The mobile clients typically render the scenes over 30 frames per second, however the current implementation of N900 is considerably slower than N93 or N95 with OpenGL ES 1.0 on certain operations such as multitexturing, performing 10–20fps. The application can run an entire day, unless the view is constantly manipulated, in which case the battery runs out in a few hours. In practice, we have not witnessed this.

The handheld platform deploys a Panasonic CF-U1 ruggedized ultra mobile PC (UMPC), extended with high quality sensors for tracking and a high quality camera. An ergonomic shell has been developed to attach and protect all the sensors [30] (see Fig. 1, right). The unit has a long battery life, and has been tested to perform continuously for up to four hours with all sensors connected and producing data. Thus, the platform's usability extends beyond simple demo sessions and supports deployment in real world situations.

Our application assumes that the user's viewpoint can be defined by tracking the camera. We rely on a GPS for location and a set of orientation sensors (inertia, magnetometer) to detect direction. The orientation sensor is attached to the camera in a way such that rotations of the camera can be estimated with minimal effort. The setup is calibrated to minimize errors; still the accuracy of tracking with sensor only methods is limited. Several methods for vision based tracking are being considered to attain finer registration. In general, these methods take the measurement of the sensors as an initial estimate and refine it by estimating changes to the video image. The description of vision based methods for tracking is out of the scope of this paper. The 3D rendering is implemented in Windows with OpenGL. The rendering speed reaches clearly interactive levels (about 40fps).

4.2 Sensor and Simulation Data

Our system leverages online availability of environmental sensor data, focusing on hydrological phenomena. HYDROSYS deploys temporary or semi-permanent sensor stations at strategic points in order to study temporal events, such as flooding caused by snow melt or leaks of poisonous substances. Typical measurements include oxygen level, conductivity, moisture of snow, temperature, water flow, solid content in water, etc.

The primary data emerging from physical sensors is sent to servers, which act as distributors. In a typical case, sensor stations send this raw sensor data using proprietary protocols (see section 6 for a sample of raw data). To manage these proprietary formats, *Wrappers* are written to parse in raw sensor data files for a GSN node to stream forward. Similarly, at the receiving end, *Virtual Sensors* are attached to the GSN node to prepare the data for local processing.

GSN utilizes the common XML-RPC scheme to communicate between nodes, where the payload is encoded with XML, and HTTP is used as the transport mechanism [18]. In this manner, any application that has the capability of utilizing raw sensor data can attach itself to the GSN, *subscribing* to the streams independently of the raw data formats or primary transmission mechanisms.

A subscriber can be, for example, a physically based on-line hydrological simulator, an alarm system, or a mobile user. However, in the last case, due to resource issues with mobile networks, on-the-fly application specific optimizations are required.

We run hydrological simulations in a separate server, which has its local preprocessed static data sets available, and receives sensor data from a GSN stream. Our simulations include for example a time-dependent water level solver implemented in Fortran. A mixed reality user can subscribe to the simulation results via the Smart Client (see next section).

4.3 Data Handling in Smart Clients

HYDROSYS mobile mixed reality applications need to be served with both static 3D content and dynamic data. As we are providing an *internal* service, we are not limited by standardized formats or services, which would be necessary for any *external* or globally available interfaces (such as GSN). Indeed, our approach is similar to that of creating Internet services at the Application Layer - for each service, a suitable protocol is developed, and the service is assigned to run in a particularly dedicated port on the host.

Our solution to the networking issue is a Smart Client, a service which inputs and processes data in various formats, from static and networked sources, delivering it in an optimized form. Large data sets, for example digital elevation maps, can be preprocessed prior to storage in a Smart Client. Streamed sensor data, entering a Smart Client via GSN, is transcoded and prepared for the mobile devices on-the-fly, caching the results for future use. Smart Client performs appropriate geo-referencing and asserts all data for a predefined projection system, directly usable in the actual application.

For the cell phone platform, the internal communication between *any* data receiving processes and a Smart Client process, both residing in the same physical host, is established through a Postgres database. Polling is avoided via *update events* with the PostgreSQL `LISTEN/NOTIFY` mechanism. We are equipped to host multiple web services and proprietary server processes in the same manner.

4.4 Mobile Networking

We have two main approaches for our mixed reality networking solutions. For our measurement campaigns with no network coverage, we arrange Wi-Fi bridges using directional antennas to support the handheld platform. The data sets for handhelds are pre-installed, and only sensor data is streamed. For cell phones, everything is sent via 3G networks. We design a protocol that encompasses 3D models and sensor data, and is tuned to suit the 3G network characteristics.

Attempts have been made to create standardized protocols intended for virtual environment applications [6], but with little success. Networked 3D computer games are in the forefront of time-critical network resource usage [28], but their networking schemes arise from very specific application needs [3,5].

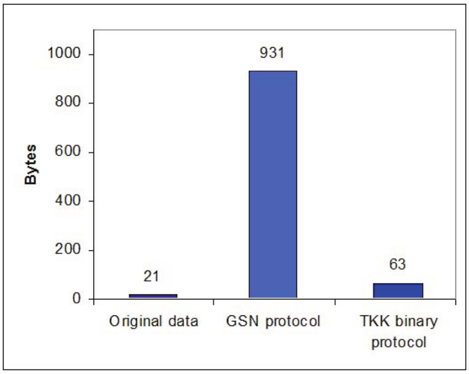


Fig. 4. Smart Client transcoding efficiency of sensor data for cell phones

Table 1. Protocol definitions for implicit content update subscriptions

```
<message name="position_update_request">
  <field name="position" type="coordinates"/>
</message>
<message name="position_update_response">
  <header>
    <container name="entities" type="entity_message" />
    <container name="removables" type="uint32_t" />
  </header>
</message>
<message name="entity_update_entities">
   <container name="entities" type="entity_message" />
</message>
<message name="entity_remove_entities">
  <container name="ids" type="uint32_t" />
</message>
```

The *M3G* standard specifies both a binary scene graph and a serialization method intended for mobile devices [1], but has no support for external acceleration schemes or application-specific features.

Mobile TCP/IP networks easily become communication bottlenecks due to increased latencies, link switches, potential congestion, unreliability and sporadically varying capacity [12,8,7]. Therefore, we would prefer to utilize the available capacity as fully as possible. Unfortunately, typical XML-RPC schemes such as SOAP do the very contrary, critically undermining our goals [15].

For our application needs, and to utilize mobile networks to the fullest, we design our own scheme with *persistent connections* running an *asynchronous protocol* with *aggregated requests*. We define our protocol with XML, but tokenize it to a compact, binary form. This mechanism has proved efficient in urban sets [21]. Table 1 presents an example of the protocol definition for dynamic updates. Here, content update subscriptions are implicitly defined by the position of the viewpoint, which is sent to server, and relevant updates received, if any. The protocol is extended as needs arise.

Figure 4 presents the efficiency of our Smart Client. A single sensor update, "`29.03.2009;08:00;12,2`" (21 bytes of data) requires 931 bytes with GSN subscription and response. Our compact scheme reduces this to 63 bytes, with the same payload. The actual binary response is only 17 bytes, *less* than the original data. Some overhead in subscriptions is caused by sending sensor station and sensor names as ASCII strings.

5 Mobile Graphics and Interaction

5.1 Cell Phone Platform

In the late 1990s, when virtual hype was at its peak, the idea of using real world based 3D models as maps in mobile devices emerged. Unfortunately, straightforward model viewing was not succesful, as graphically rich models exceeded the capabilities of the devices. Even when the models were drastically simplified, rendering speed was far below interactive rates [24]. Although it may appear that the issue was in insufficient hardware, the problem actually lies in the fact that the resources of *any* hardware are limited, while a 3D model can, in principle, span infinitely large areas with infinitely accurate details. To overcome the resource bottleneck, we assume that our model will never completely fit to our memory, and resort to *out-of-core rendering*, where only a small subset of the scene is held in memory, with only the currently necessary levels of detail.

We have previously developed a mobile 3D map system for urban environments, which is based on pre-calculated visibility lists, useful in heavily occluded environments. In HYDROSYS, our environment is semiurban, with both natural and manmade structures, with less occluding features. We therefore adapt our visibility scheme to the characteristics of the environment by two passes, a density pass and the actual, density-modified visibility computation pass.

For level-of-detail management, we have taken into use an appearance-preserving simplification method, where we replace geometry with textures [9]. We also apply texturing techniques to emphasize scene topography [10], and support translucency. Figure 5 presents the use of *lightmaps* to slightly emphasize topographical features.

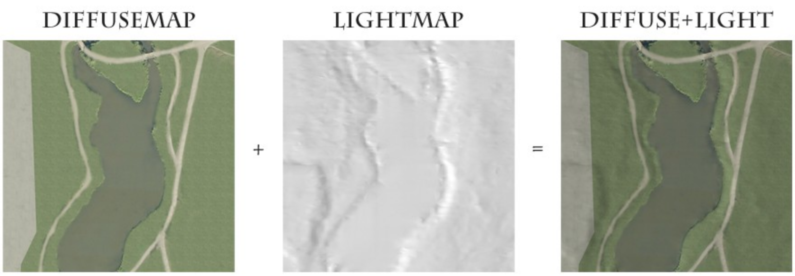


Fig. 5. A pre-calculated lightmap emphasizes scene topography, avoiding per-vertex lighting calculations and allowing coarser geometry

Our citizen scenario required a direct, sensor data based modification of the appearance of the environment. For this end, we create a segmented terrain model, where each visually separable feature is tessellated into its own, unique mesh. Semantically similar neighboring meshes, such as river surface patches, are connected onto a topology. This method separates our system from the traditional layer-based GIS approach.

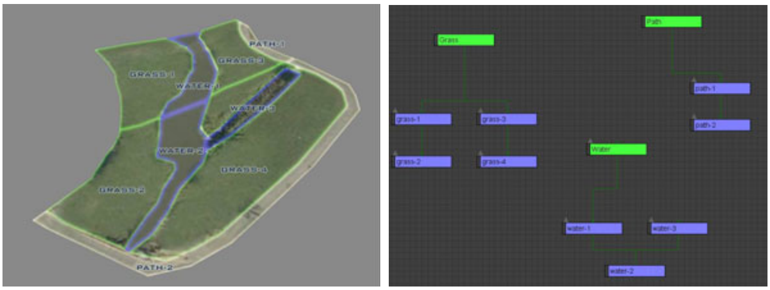


Fig. 6. Model segmentation (left) with related connectivity graphs (right)

Figure 6 presents our mesh graph. The terrain (left) is split to meshes, and hierarchies and connections built in a 3D modeler. Our pre-process then creates an optimized binary model, where the most accurate level of detail is gradually replaced with coarser geometry, rasterizing the original geometry. The pre-process also creates lightmaps for emphasizing the topography.

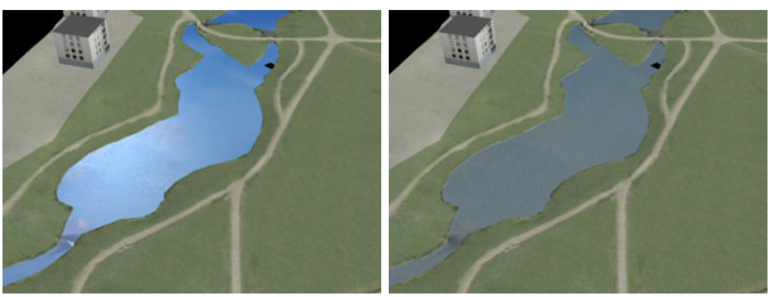


Fig. 7. Two sensor data visualizations, where *turbidity* modulates the look of a pond

Figure 7 presents sample visualizations of a pond from our case area, where we have run measurement campaigns. When the measured *turbidity* is only light, we visualize the pond with high reflectivity. With heavy turbidity, we apply a greyish texture. The sensor measurement directly affects a factor between an environment map depicting a sky, and a "murky" diffuse texture. The presented visualization is aimed at the public. For professionals, we also provide conventional temporal sensor value plots (Figure 2, right).

With our segmented model, annotations and sensor data sets are connected to the underlying data structures (meshes, topologies and hierarchies). Therefore, they are accessible directly from the environment, or via the traditional point-based billboard icons. Figure 8 presents the pond with a sensor station attached, and three public annotations. In the screenshot, the pond is *selected*, depicted by a dark emphasis. A selection pops up a context-sensitive menu, displaying the available sensor measurements, annotations and online discussions, and allows placement of new (photo) annotations and even sensor stations.

Our system achieves scalability by limiting updates to events that are of interest to the user. As an advanced feature, we use an implicit mechanism to define *interest*. For example, if the user is observing a graph of a sensor value, that graph is automatically updated as new data emerges. Similarly, only updates that happen in users' current view are transmitted. This is possible due to shared visibility tables between a server and its clients. As the user maneuvers in the 3D view, simple view position updates are sent to the server, which then knows what the user can potentially observe (see Table 1).

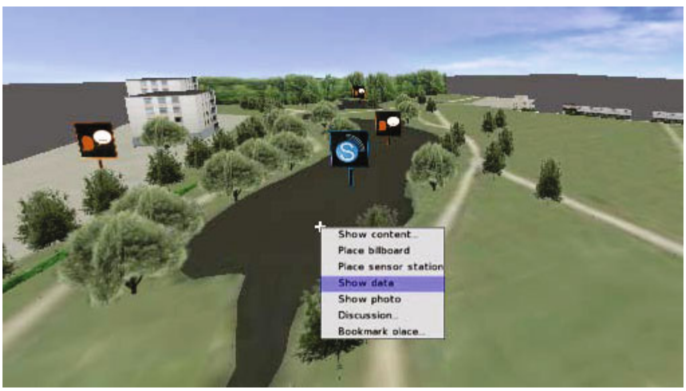


Fig. 8. The Hydrosys mobile 3D interface. The online sensor measurements and annotations are available from the icons or via the scene itself.

5.2 Handheld Platform

The handheld platform promotes the usage of the various kinds of visualizations by supporting smooth transitions between multivariate sensor data: 1D, 2D and 3D visualizations including AR are tightly interwoven by (a) making available all kinds of visualization by default, (b) allowing to quickly select the various kinds of visualizations using intuitive user interfaces, and (c) being able to compare the results of different kinds of visualization. Thereby, data can be shown in its pure form as tabular data, but also registered to the environment by using the location of the sensor station the data is obtained from. This kind of visualization is generally called a "label" and appears as geo-referenced text box in the 3D view of the user. Users can also inspect the temporal development of data by viewing plots. These plots are pre-processed on the sensor network. Depending on the refresh rate of the sensors, plots can be updated rapidly. Finally, users can make use of various kinds of overlays: Overlays are geo-referenced textures overlaid on top of the digital terrain model. When combined with video footage, these overlays appear to be laid on top of the environment itself. Here, accurate registration through correct tracking of the camera (hence, the user's viewpoint) is needed. The registration process of 3D visualization (and that of AR as well) requires tracking of the viewpoint and the presentation of data in 3D, with respect to the viewpoint. Using hybrid-tracking methods, HYDROSYS is trying to avoid discrepancies between the video and the overlaid data. Implicitly, all data that is to be presented in 3D must be geo-referenced and in the same system of projection. The 3D presentation, user interfaces and visualizations must be carefully planned to cope with perceptual issues arising from reduced screen space, uncontrolled lighting conditions and other factors inherent to outdoor scenarios [17]. Finally, a mobile user relies and depends upon other local and remote users, and requires means to communicate and collaborate with them.

Overlays can have multiple origins. Within HYDROSYS, several services generate a multitude of different 3D data sets, varying from simple thermal interpolations up to more complex simulations, for example those that are used for interpreting the level of danger of avalanches. Furthermore, overlays are available that refer to non-simulation data, such as network coverage in a particular area. The merging of numeric data through labels and the graphical simulation outcomes via overlays allows users to directly combine multiple kinds of visualizations in a single view.

To aid depth perception, one of the most common perceptual issues in an augmented reality, our 3D visualization includes the possibility to show contour lines. Contour lines are line-based visualizations overlaid over the DTM that provide in particular depth cues and communicate very well the structure of the environment, and can be color-coded, for example representing temperature differences (isotherms).

One particularity of the handheld system is the actual number of cameras being deployed in the field, including users' own handheld units and potentially moving overhead footage taken from a blimp. If multiple cameras are dispersed throughout the environment, users can observe the site from multiple perspectives. Observing the environment from different viewpoints, is expected to give users a better overview of the situation, allowing them to improve the understanding of the spatiotemporal characteristics and development of environment processes in their actual context. Nonetheless, accessing these different perspectives comes at its own cost in spatial understanding, in particular when accessing remote views the viewer can easily confuse spatial cues between views. Based on a series of studies, we have developed several techniques to aid spatial understanding of the environment from multiple perspectives [29]. It thus also provides a common baseline between users to improve collaboration. Similar to the cell

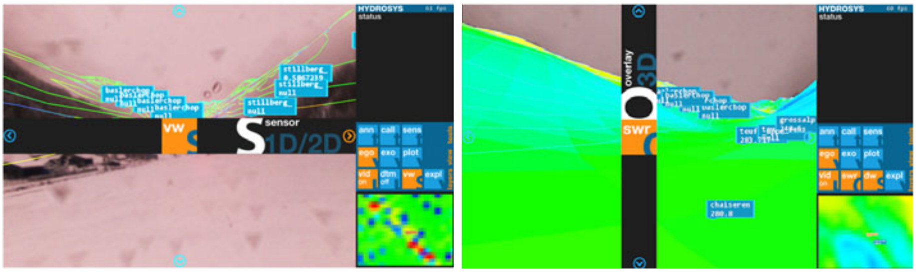


Fig. 9. The handheld AR view. Sensor data selection (left) and overlay selection (right). Most of the view is filled by snow and sky. Small buildings are visible on the left.

phone system, users may also add annotations to note down findings or ideas, further extending the collaborative aspects of the system.

6 Application Feedback

The kind of visualization the user prefers to use is both depending on the task at hand, and, to a lesser extent, also on the users previous experience and stance towards various visualization methods. As was confirmed during our user-centered design process, the field of mobile GIS is in transition: whereas the majority of users still is accustomed to the more classical 1D and 2D representations, a clear but slow movement towards 3D visualization can be observed. Hereby, the opinion on 3D visualization varies widely from users that avoid those formats, claiming them to be a mere "fancy thing", up to users that actively deploy 3D visualizations in their daily routine.

As we can draw from previous experience, the opinion on new visualization methods tends to change over time. This is yet another good reason for our UCD approach: by keeping end-users involved from start till end of the project, research prototypes can be tested at every stage. Frequently, users get more interested in using 3D visualizations once the stage of development has left the initial prototypical stage. Hereby, the competitive advantage can also be found in the inclusion of simulation results - though not discussed in detail in this paper, the simulation pipeline is a great improvement over previous work principles, by closely integrating data access with simulation specification, and interpretation procedures for visualization purposes. The mixed reality user interfaces sometimes surprise the users who are not accustomed to such applications. The AR metaphor has been relatively easy to assimilate, but 3D maneuvering on a cell phone has shown to be a new experience. Here we are proceeding towards similar sensor based maneuvering as with the handheld (for example, rotating the device rotates the 3D view), which has appeared more pleasing in our trials. Technically, both platforms have performed adequately. Regarding the visualizations, professionals still like their graphs and absolute values.

Finally, our application provides, in addition to sensor data modified visualizations of the environment, online location and entity based discussion forums.

Fig. 10. A semipermanent sensor station (left) and end users enjoying the live data feed at Kylmäoja, Finland (right)

Any user can place an annotation with a photo, which can act as a discussion topic. This provides a basic collaboration tool also for professionals, although our end users considered this to be mainly a feature for the public.

7 Conclusions

HYDROSYS makes several important contributions to the field of mobile applications used for monitoring and managing environmental processes: the system optimizes streaming of sensor data, includes highly graphical mixed reality representations embedding multivariate sensor data visualizations, promotes access to near real time simulations and other physically-based computations, and supports semantically meaningful public participation. We have run several campaigns with semipermanent sensor stations, involving end users in the field (Fig. 10). We are currently developing mobile sensors that can be carried along to lower the threshold of performing quantitative environmental measurements and to speed up reaction times to catch near instantaneous events.

Acknowledgment. This work was made possible with the EU FP7/DG-INFSO project 224416 Advanced Spatial Analysis Tools For On-site Environmental Monitoring And Management (HYDROSYS).

References

1. Aarnio, T.: Jsr 184: Mobile 3d graphics api for j2me (2005), `http://www.jcp.org/en/jsr/detail?id=184`
2. Aberer, K., Hauswirth, M., Salehi, A.: A middleware for fast and flexible sensor network deployment. In: VLDB 2006: Proceedings of the 32nd International Conference on Very Large Data Bases. pp. 1199–1202. VLDB Endowment (2006)
3. Abrash, M.: Ramblings in Realtime. Dr. Dobb's Sourcebook (2000)
4. Azuma, R., Baillot, Y., Behringer, R., Feiner, S., Julier, S., MacIntyre, B.: Recent advances in augmented reality. IEEE Computer Graphics and Applications 21(6), 34–47 (2001), `citeseer.ist.psu.edu/azuma01recent.html`

5. Bonham, S., Grossman, D., Portnoy, W., Tam, K.: Quake: An example multiuser network application - problems and solutions in distributed interactive simulations. Technical report, University of Washington, cSE 561 Term Project Report (May 2000)
6. Brutzman, D.P., Zyda, M., Watsen, K., Macedonia, M.R.: Virtual reality transfer protocol (vrtp) design rationale. In: WET-ICE 1997: Proceedings of the 6th Workshop on Enabling Technologies on Infrastructure for Collaborative Enterprises, pp. 179–186. IEEE Computer Society, Washington (1997)
7. Chakravorty, R., Cartwright, J., Pratt, I.: Practical experience with tcp over gprs. In: GLOBECOM 2002, vol. 2, pp. 1678–1682. IEEE, Los Alamitos (2002)
8. Chan, M.C., Ramjee, R.: Tcp/ip performance over 3g wireless links with rate and delay variation. Wirel. Netw. 11(1-2), 81–97 (2005)
9. Cohen, J., Olano, M., Manocha, D.: Appearance-preserving simplification. In: SIGGRAPH 1998: Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques, pp. 115–122. ACM, New York (1998)
10. Döllner, J., Baumann, K., Hinrichs, K.: Texturing techniques for terrain visualization. In: VISUALIZATION 2000: Proceedings of the 11th IEEE Visualization 2000 Conference (VIS 2000), pp. 227–234. IEEE Computer Society, Washington (2000)
11. Feiner, S., MacIntyre, B., Hollerer, T., Webster, A.: A touring machine: Prototyping 3d mobile augmented reality systems for exploring the urban environment. In: ISWC 2000, p. 74 (1997)
12. Holland, G., Vaidya, N.: Analysis of tcp performance over mobile ad hoc networks. Wirel. Netw. 8(2/3), 275–288 (2002)
13. Holtzblatt, K., Wendell, J.B., Wood, S.: Rapid Contextual Design: A How-To Guide to Key Techniques for User-Centered Design. Morgan Kaufmann, San Francisco (2004)
14. Jones, M.T.: Google's geospatial organizing principle. IEEE Computer Graphics and Applications 27(4), 8–13 (2007)
15. Kangasharju, J., Tarkoma, S., Raatikainen, K.: Comparing soap performance for various encodings, protocols, and connections. In: Conti, M., Giordano, S., Gregori, E., Olariu, S. (eds.) PWC 2003. LNCS, vol. 2775, pp. 397–406. Springer, Heidelberg (2003)
16. King, G.R., Piekarski, W., Thomas, B.H.: ARVino - Outdoor augmented reality visualisation of viticulture GIS data. In: Werner, B. (ed.) Proceedings of the 4th IEEEACM International Symposium on Mixed and Augmented Reality, IEEE/ACM/IEEE Comp Soc; XSENS; NOKIA; Trivisio; SIEMENS; Engn. Syst. Technol.; INTERSENSE; Fdn Stift, pp. 52–55. IEEE Computer Society, Los Alamitos (2005), `http://portal.acm.org/citation.cfm?id=1104996.1105174`
17. Kruijff, E., Swan, E., Feiner, S.: Perceptual Issues for Augmented Reality Revisited. In: Proceedings of the 9th IEEE/ACM International Symposium on Mixed and Augmented Reality, ISMAR 2010 (to appear, 2010)
18. Laurent, S.S., Johnston, J., Dumbill, E.: Programming Web Services with XML-RPC, 1st edn. O'Reilly, Sebastopol (2001)
19. Milgram, P., Kishino, F.: A taxonomy of mixed reality visual displays. IEICE Transactions on Information Systems E77-D(12), 29–33 (1994)
20. Norman, D., Draper, S.W. (eds.): User Centered System Design. Lawrence Erlbaum Associates, Mahwah (1986)
21. Nurminen, A.: Mobile, hardware-accelerated urban 3d maps in 3g networks. In: Web3D 2007: Proceedings of the Twelfth International Conference on 3D Web Technology, pp. 7–16. ACM Press, New York (2007)

22. Nurminen, A.: Mobile 3d city maps. IEEE Computer Graphics and Applications 28(4), 20–31 (2008)
23. Piekarski, W., Thomas, B.H.: Tinmith-evo5 – An Architecture for Supporting Mobile Augmented Reality Environments. In: IEEE and ACM International Symposium on Augmented Reality, ISAR 2001, p. 177 (2001)
24. Rakkolainen, I., Timmerheid, J., Vainio, T.: A 3d city info for mobile users. Computers and Graphics 25(4), 619–625 (2001)
25. Reitmayr, G., Drummond, T.W.: Going out: Robust model-based tracking for outdoor augmented reality. In: IEEE/ACM International Symposium on Mixed and Augmented Reality, ISMAR 2006 (2006)
26. Schall, G., Mendez, E., Kruijff, E., Veas, E., Junghanns, S., Reitinger, B., Schmalstieg, D.: Handheld Augmented Reality for underground infrastructure visualization. Personal and Ubiquitous Computing 13(4), 281–291 (2008), `http://www.springerlink.com/index/10.1007/s00779-008-0204-5`
27. Schall, G., Mendez, E., Kruijff, E., Veas, E., Junghanns, S., Reitinger, B., Schmalstieg, D.: Handheld augmented reality for underground infrastructure visualization. Personal Ubiquitous Comput. 13(4), 281–291 (2009)
28. Smed, J., Kaukoranta, T., Hakonen, H.: Aspects of networking in multiplayer computer games. The Electronic Library 20(2), 87–97 (2002)
29. Veas, E., Mulloni, A., Kruijff, E., Regenbrecht, H., Schmalstieg, D.: Techniques for View Transition in Multi-Camera Outdoor Environments. In: Proc. of Graphics Interface 2010, GI 2010 (2010)
30. Veas, E., Kruijff, E.: Vesp'R: design and evaluation of a handheld AR device. In: ISMAR 2008: Proceedings of the 7th IEEE/ACM International Symposium on Mixed and Augmented Reality, pp. 43–52. IEEE Computer Society, Washington (2008)
31. Wagner, D., Schmalstieg, D.: First steps towards handheld augmented reality. In: IEEE International Symposium on Wearable Computers (ISWC), p. 127 (2003), `http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.68.7550\&rep=rep1\&type=pdf`

Inferring and Focusing Areas of Interest from GPS Traces

Pablo Martinez Lerin, Daisuke Yamamoto, and Naohisa Takahashi

Dept. of Computer Science and Engineering, Nagoya Institute of Technology,
Gokisocho, Showaku, Nagoya 466-8555 Japan
{pablo,daisuke,naohisa}@moss.elcom.nitech.ac.jp

Abstract. Advanced GIS applications and GPS loggers allow travelers to record their own tracks and later examine, modify and share them. Searching for the areas of interest, AOIs, however, is often not an easy task, especially with long routes. This paper proposes a system for inferring and focusing AOIs, from a GPS trace. The proposed system consists of three main functions: fragmentation, defragmentation, and focusing. The fragmentation detects the changes of the travelling pace and decomposes the GPS trace into a large number of small fragments according the traveling pace while the defragmentation composes adjacent fragments into one fragment which is inside the AOI. The focusing provides a Focus+Context+Glue map where the Focus is an area of a large-scale map including the AOI that enables users to understand details of the AOI. We have developed a prototype of the proposed method that includes the above features and evaluated the feasibility and advantages of the proposed system.

Keywords: GIS, GPS, Web mapping, Focus+Glue+Context, Travel record, Travel behavior, Spatio-temporal data mining.

1 Introduction

The increasing development of GIS products, such as GPS loggers, and GIS applications encourages travelers to record their own tracks. Nowadays many GIS applications are accessible, especially web mapping applications for individual use, that allow users to easily share, analyze and add information to their travel records.

Users usually load and display the GPS trace of a travel in a GIS application to perform basic functions such as examine and add contents in some interesting areas of the travel. The areas of the travel the user is interested in are referred in this paper as areas of interest (AOIs). Sometimes it may be a difficult task to search where the AOIs of a travel are. It is particularly a problem in two situations: (1) when the travel was made long time ago so it is difficult to remember what happened and where and (2) when the travel was not made by the user, for example it when it is shared. Searching for the areas by using successively the map functions scroll, zoom in and zoom out is really a though and unpleasant task, especially in long trips. Finding the AOIs of a travel is therefore a very important task.

In previous studies, we have proposed an advanced Focus+Glue+Context map system EMMA to enhance the map visualization in GIS applications [1,2]. We have

K. Tanaka, P. Fröhlich, and K.-S. Kim (Eds.): W2GIS 2011, LNCS 6574, pp. 176–187, 2011.

applied the EMMA system to a travel route editor in a recent study [3]. In this paper, we propose a system for inferring AOIs from a GPS trace and focusing the inferred AOIs by using EMMA.

We have observed that usually the travelling pace is slower inside the AOIs than outside the AOI. The main key of our approach is to infer the travelling paces used on a travel by using a probability density function. It is an important challenge since travel records can be very variant and different. On the one hand, the way of travelling (by walk, bike, car …) may be different from travel to travel and, on the other hand, the travelling speed may be different from user to user.

2 Related Work

Several studies deal with the inference of information from spatio-temporal data, mostly GPS trace data.

Many of those studies analyze large amounts of floating car data (FCD) and try to find patterns (spatio-temporal pattern mining). Schilich and Axhausen introduce and compare different methods to measure similarity of travel behavior [4]. Zhu proposes an Automatic Incident Detection, AID, approach based on FCD analysis [5]. Yue analyses FCD from many taxis in a big area, for example a city [6]. Then, by using point pattern analysis (PPA) methods the system analyses the distribution of the density of taxi pick-up and drop-off points and it infers the hot spots of the city.

On the other hand, other studies propose systems for individual use that try to derive information from a single GPS trace data (spatio-temporal data mining). Wolf proposed a GIS that analyzes a GPS trace data and derives the places where the user stopped and the purpose of each stop [7]. The inference is based on the location, the duration of the stop and the time of the day. Ashbrook and Starner system analyzed GPS data taken from many users over an extended period of time and inferred meaningful locations at multiple scales [8]. These locations are used to build a Markov model than can be used in order to predict the user movements.

Our proposed system automatically infers the AOIs from a GPS trace by using the speed, location and time of the GPS points. The proposed system, which is for individual use and derives information from a single GPS trace, is different from the previous works mainly because it infers important areas from a GPS trace, as opposite to infer important points or locations. We consider that finding the points where the user stopped may not be enough information. In some AOIs the travelers may not stop and for example if they go to a shopping street they may want to know not only where they stopped but also the boundaries of the shopping area they visited.

3 Observations

An area of interest, AOI, is an area of the travel where the traveler performs an activity or feels a strong impression or impact. For example go shopping, rest, assist to an event, contemplate a sightseeing spot and meet with friends.

When analyzing the traveler's movement inside and outside AOIs, we can observe the following behaviors: (B1) Travelers move slower inside AOIs than outside AOIs in many cases. (B2) Travelers stay for a relative long time inside an AOI in many cases. (B3) In few cases travelers move inside AOIs as fast as outside AOIs.

Table 1. A set of GPS trace data used for the observation and the experiments in this paper

GPS trace data	Transportation	Length (km)	Number of points	Time duration (hours)	Average speed (km/h)	sampling interval (sec)
T1	car, walk	40	2387	6	12	5
T2	car, walk	76	4887	7	11	5
T3	car, walk	57	3682	5	11	5
T4	car, walk	116	3303	5	25	5
T5	car, walk	60	4392	6	10	5
T6	car, walk	53	2448	4	16	5
T7	car, walk	277	5460	9	26	5
T8	train, bus, walk	28	2824	6	7	5
T9	bus, walk	28	3616	10	5	5
T10	subway, walk	13	1035	4	6	5
T11	train, bus, walk	100	19247	8	17	1
T12	walk	5	4163	1	4	1

By using a GPS logger during a travel, the trace of the travel can be saved and later analyzed. A GPS logger simply logs at a regular time the current track point information: usually the current position (latitude and longitude) of the device, the time when it is recorded and the travelling speed.

A GPS trace data T is a sequence of n track points $p_i(i=1,..n)$ described as follows: $T = [p_1, p_2, \ldots, p_n]$

A track point p_i is a 4-tuple of (timestamp, speed, latitude, longitude) and each element of the tuple is referred to by using the expression like p.speed.

We have analyzed the GPS trace data of 12 real travels, described in the Table 1. We have observed the following characteristic in the GPS data: (C1) the travelling speed is not uniform even when the traveler tries to move at a constant pace.

Since when reaching AOIs the travelling pace usually changes, we are interested in extracting the travelling paces used on a travel. In order to analyze the speed of the trace points and infer the different travelling paces used, first we introduce a cumulative distribution function F(s) for the traveler's speed s as follows.

$$F(s) = \frac{|\{p \mid p.\text{speed} < s, p \in T\}|}{|T|}, \text{ where } |S| \text{ is the numbers of elements in a set S.} \quad (1)$$

Then, we define a probability density function f(s) for the traveler's speed s :

$$f(s) = \frac{d(F(s))}{ds} = \frac{|\{p|\ s - \Delta s < p.speed \le s, p \in T\}|}{|T|} \tag{2}$$

Figure 1 shows the functions F(s) and f(s) which are calculated by using Δs=0.25km/h for the trace T8. Above 20 km/h the values keep the same tendency and are not relevant. Note that GPS loggers save very few points during the fast paces.

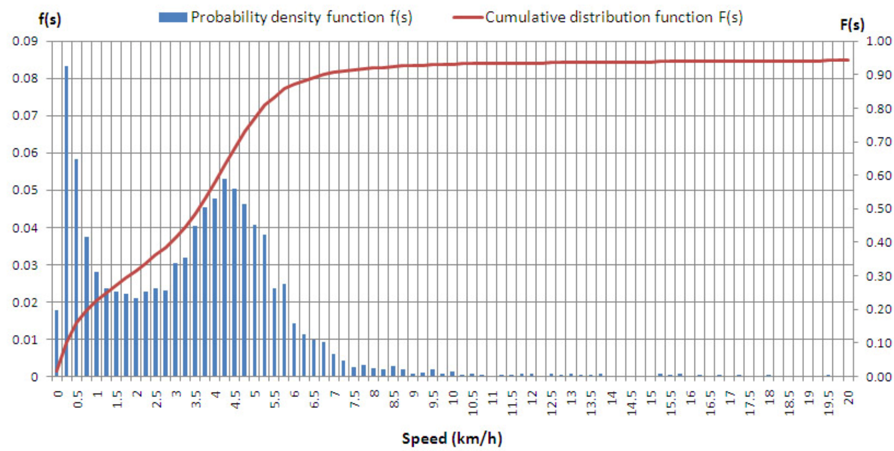


Fig. 1. Functions F(s) and f(s) with Δs=0.25km/h and speed s from 0 to 20km/h

By analyzing the GPS data of a travel by using the probability density function f(s) for the traveler's speed s, we can observe the following characteristics:

(C2) We can find peaks in the graph. It is inferred that the peaks found on the graph represent the different slow paces used on the travel.

(C3) From the graph we can infer speed thresholds that separate the different travelling paces used.

(C4) Depending on the GPS data and the Δs value used in the function f(s), there may appear small peaks that does not represent a travelling pace.

4 Proposed System

4.1 Overview

This section presents a system for automatically inferring and focusing AOIs from the GPS trace of a travel. In this section we refer to the results (B1-2)(C1-4) described in the previous section. The proposed system deals with the observations B1 and B2 but not B3, i.e. the proposed system infers the areas where the traveler moved slowly in a relatively long duration. The system consists of the following three steps.

STEP1 (fragmentation): It analyzes the speed at the points of a GPS trace and regarding the observation (B1) decomposes the trace into a large number of small fragments.

STEP2 (defragmentation): It enhances the fragmentation regarding the observation (B2) and the GPS data characteristic (C1).

STEP3 (AOI extraction and focusing): It extracts AOIs from the GPS trace fragments. By using a Focus+Glue+Context map it provides two advanced functions to automatically focus and display an AOI.

The above three steps are explained in detail in the following subsections.

4.2 Fragmentation

The purpose of the fragmentation step is to decompose the GPS trace in fragments based on the slow travelling paces used on the travel. As the observations (C2) and (C3) show, speed thresholds that distinguish the paces can be inferred by analyzing the probability density function f(s) for the traveler's speed s. To avoid the problem caused by the (C4), we introduce a new function f'(s) that smoothes the function f(s).

$$f'(s) = \frac{f(s-\Delta s) + f(s) + f(s+\Delta s)}{3} \tag{3}$$

We consider that the classification of the paces into three classes (very slow, slow and fast) provides enough information to infer the AOIs of a travel. Many classes would result intrusive and difficult to understand. Therefore, the fragmentation step uses two speed thresholds in order to distinguish the three classes of travelling paces.

The fragmentation is achieved by the following procedure. Its inputs are a GPS trace data T, the function f'(s) and the function F(s) and its outputs are two sequences of fragments Q1 and Q2 treated as queues.

Definitions used on the fragmentation and defragmentation processes:

A track point p is composed by three parameters concerning the moment the point was saved: time, location and speed.

A GPS trace data T is a sequence of n track points pi(i=1,..n) described as follows: T = [p1,p2,…,pn]. The parameters of the points can be accessed using the dot operator, for example: T[3].speed .

A fragment f is composed by three parameters: start (the start point of the fragment), end (the end point of the fragment) and distance (the distance from the previous fragment in meters). The two points, start and end, are track points from T. The parameters of the start and end point can be accessed using the point operator, for example: f.start.time .

Functions used on the fragmentation and defragmentation processes:

enqueue(Q,e)

It adds the element e to the end of the queue Q.

e = *dequeue*(Q)
It removes the first element e in the queue Q and returns it.

d = *dist*(T[i],T[j])
It recursively calculates the distance d between the points T[i] and T[j] as follows.

$$d = \begin{cases} 0 & \text{if } j \leq i \\ \text{Euclidian distance between T[i].location and T[j].location} & \text{if } j=i+1 \\ dist(T[i],T[i+1])+dist(T[i+1],T[i+2])+\ldots+dist(T[j-1],T[j]) & \text{otherwise} \end{cases}$$

(fr,j+1) = *make-fragment*(T,i,s,dp)
Starting from the point T[i], it searches the first point T[j+1] in T which speed is higher than s and uses the input parameters to create a new fragment fr as follows.
fr.distance=dp,
fr.start =i,
fr.end = j,
Where
$j \geq i$,
T[k].speed ≤ s for $i \leq k \leq j$,
T[j+1].speed > s or $j = |T|$

(dq,j) = *find-next-fragment*(T,i,s)
Starting from the point T[i], it searches the first point T[j] in T which speed is less or equal than s and returns the value j and the distance from T[i] to T[j] as follows.
dq = *dist*(T[i],T[j]),
Where
$j \geq i$,
T[k].speed > s for $i < k < j$,
T[k].speed ≤ s for $k = j$
If the end of T is reached and T[j] is not found, it returns dq=0 and j=|T|+1 .

Q = *make-seq*(T,s)
It searches all the fragments of the trace T where the speed is less or equal than s and return the fragments in a queue Q as follows. Figure 2 shows an example.
If s = null, returns Q= null,
Otherwise,
Initialize Q
(dq, i) = *find-next-fragment* (T,1,s)
While (i ≤ |T|) do
(f1,i) = *make-fragment*(T,i,s,dq),
enqueue(Q,fi)
(dq, i) = *find-next-fragment* (T,i,s)

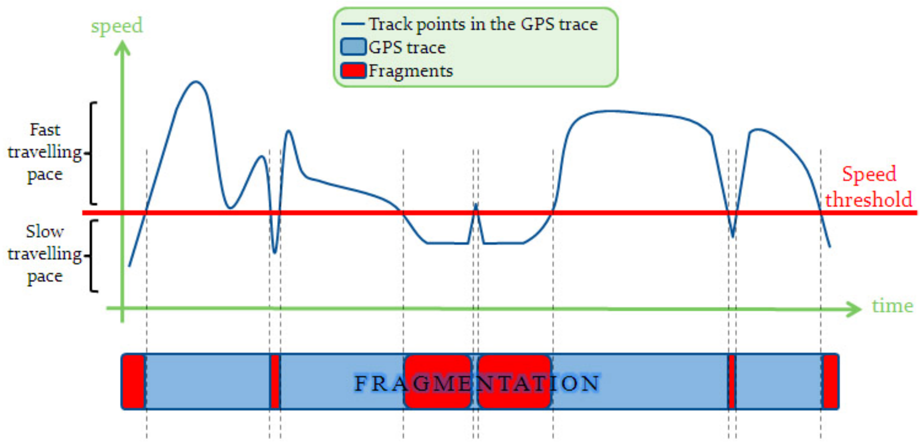


Fig. 2. Example of the *make-seq* function

The fragmentation process follows the next procedure:

STEP1-1. It builds the f'(s), by smoothing the probability density function f(s) of the traveler's speed. The functions f(s) and f'(s) are defined in (2) and (3) respectively.

STEP1-2. It detects several local minimums from the f'(s). It also detects the saturation point of the F(s) and regards the point as a local minimum. This is because we cannot find a local maximum and local minimum when the speed is very fast. The local minimums are speed threshold values that distinguish the travelling paces. It sorts the local minimums in ascending order and names M[1],M[2],…,M[k].

STEP1-3. It selects two speed-thresholds ST[1] and ST[2] from M[1] to M[k] detected in the STEP1-2 based on the following rules.
[rule1] It discards M[j] if F(M[j]) > 0.7
F(s) represents the ratio of points selected by the threshold s. The reason for rule1 is that a threshold that selects almost all the points in T does not help the AOI inference.
[rule2] ST[1] is the smallest and ST[2] is the largest in the local minimums which are not discarded by the rule1. If only one local minimum remains, ST[2] becomes null.

STEP1-4 It composes two queues of fragments, Q1 and Q2, by gathering consecutive points from T based on the points' speeds as follows.

Q1 = *make-seq*(T,ST[1])

Q2 = *make-seq*(T,ST[2])

Q1 and Q2 are the result of the fragmentation process.

4.3 Defragmentation

The following problem is found after the fragmentation process: sometimes an inferred inter-travelling-pace transition is actually an intra-travelling-pace change. The defragmentation step attempts to fix the problem, which occurs in two situations:

(1) when between the fragments there is a short gap, it may be caused by anomalous speed values (GPS noise), and (2) when between the gaps there is a short fragment, it may be caused when the car stops a short time in a traffic light.

The defragmentation is achieved by the following procedure. Its inputs are two sequences of fragments Q1 and Q2, received from the first step, and its outputs are two new sequences of fragments NS1 and NS2.

Besides the definitions and functions described in the previous subsection, the following functions are used on the defragmentation process:

push(S,e)
It adds the element e to the top of the stack S.

e = *pop*(S)
It removes the top element e in the stack S and returns it.

NS = *merge-neighboring-fragments*(Q)
It returns in a stack NS the fragments from the queue Q merging consecutive fragments that are nearer than a specified distance, , as follows.

If Q = null, returns NS= null,
Otherwise,
Initialize NS
push(NS, *dequeue*(Q))
While there are fragments in Q:
fb= *dequeue*(Q)
If fb.distance < ,
fa=*pop*(NS)
fc.start = fa.start
fc.end = fb.end
push(NQ,fc)
Otherwise,
push(NQ,fb)

NS = *remove-small-fragments*(S)
It returns in the sequence NS the fragments from the sequence S with duration higher than τ as follows.

If S = null, returns NS= null,
Otherwise,
Initialize NS
For each fragment f in S:
if f.end.time – f.start.time > τ, f is added to NS.

We use 20 meters and 5 minutes for δ and τ, which were determined based on the results of the preliminary experiments.

The defragmentation process follows the next procedure:

STEP2-1. It merges neighboring fragments as follows.
NQ1 = *merge-neighboring-fragments*(Q1)

NQ2 = *merge-neighboring-fragments*(Q2)

STEP2-2. It deletes small fragments as follows.

NS1 = *remove-small-fragments*(NQ1)

NS2 = *remove-small-fragments*(NQ2)

NS1 and NS2 are the result of the defragmentation process.

4.4 Areas of Interest Extraction and Focusing

The fragments returned by the defragmentation step are used in this step in order to infer and focus the AOIs of the travel. The trace of a fragment is defined as the trace formed by all the points in T from the fragment start point to the fragment end point.

Since the fragments represent places of the travel where slow travelling paces were used, an AOI is extracted from each fragment of the travel. An AOI is a circular area over the map composed of three attributes: center (latitude, longitude), diameter (meters) and level (yellow or red). An AOI extracted from a fragment is defined as the smallest circular area that includes all the trace of the fragment. Besides, the level of an AOIs is red when extracted from a fragment included in NS1 and yellow when extracted from a fragment included in NS2. Over the map screen, the traces of the fragments from NS1 are drawn in red, the traces of the fragments from NS2 are drawn in yellow, and the rest of the GPS trace is drawn in green.

Fig. 3. Results of the experiment 1 using GPS trace data shown in Table 1

Note that since the speed threshold used to obtain the fragments in NS1 is always lower than the speed threshold used in NS2, the fragments in NS1 are always included inside the NS2 fragments. In other words, the red fragments are a detailed fragmentation of the yellow fragments. Figure 3 shows the inference results for the GPS trace T1, described in the Table1, and the importance of using two different kinds of slow paces (two different levels of AOIs).

By using the advanced Focus+Glue+Context map system EMMA (Elastic Mobile MAp)[1,2], the proposed system provides the users with two functions associated to each inferred AOI: (1) the focus function and (2) the detailed focus function.

The focus function displays the whole travel route as big as possible and displays in detail the selected AOI over the same screen. Users can easily examine the AOI and at the same time understand the geographical relations with the rest of the route. Figure 3 shows the effect of using the function in the AOI number 2.

The detailed focus function displays the AOI in the most detailed map scale and the surroundings of the AOI in a larger map scale. Users can quickly access an AOI and clearly identify all the details of the AOI and the map such as the map labels.

5 Evaluation

A prototype system has been developed that implements the proposed system. The prototype is a Flash application available through internet in common web browsers. A demo movie of the prototype is also provided[1]. Additional useful functions have been implemented, for example a function to export all the information of the inferred AOIs in the geographical file format kml, which is an international standard [9]. That way the inference can be stored and load in any compatible GIS application, such as Google Earth [10]. A snapshot of the prototype system is shown in Figure 3.

We have compared the AOIs provided by the travelers, according to their memory, with the AOIs inferred by the proposed system in order to evaluate the goodness of the inference. We use two metrics, recall and precision, defined as follows:

At = A set of AOIs provided by the traveler, the real AOIs.
As = A set of AOIs inferred by the proposed system.
Recall = $|At \cap As| / |At|$
Precision = $|At \cap As| / |As|$

All of the GPS traces in Table 1 were used in this experiment. Table 2 shows the numbers of AOIs in the sets At and As and the number of AOIs in their intersection $At \cap As$. Figure 4 shows the bar-chars of a pair of precision and recall.

Table 2. Relation of the sets At and As

GPS trace data	T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12
$\|At\|$	5	6	3	3	6	4	9	5	7	2	3	2
$\|As\|$	9	5	5	3	8	6	9	7	7	4	5	2
$\|At \cap As\|$	5	3	3	3	6	4	8	4	6	2	3	1

[1] http://tk-www.elcom.nitech.ac.jp/demo/fisheye.html

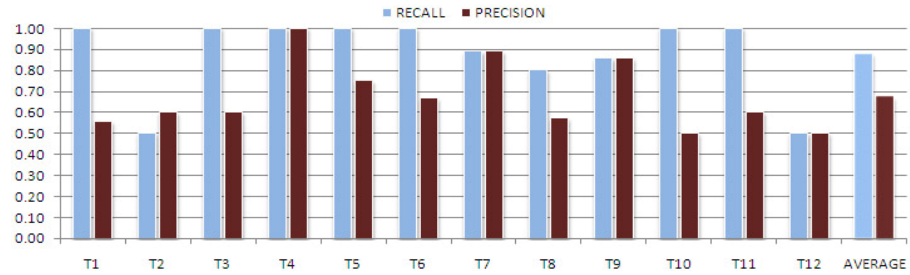


Fig. 4. Bar-chars of precision and recall for each GPS trace data evaluated and average values

From Figure 4, we can see that the recalls are quite good for many traces; however, we found that there are some cases where some of the AOIs provided by the traveler were not inferred in this experiment. We now discuss the reasons. On the one hand, in many of the travels evaluated (T2, T7 and T8), it was not a complete error because the AOIs not inferred where inside a big inferred AOI. For example, the system inferred an AOI including a park but not one of the AOIs inside the park. Therefore, just by searching inside the big AOIs the user can find the AOIs not inferred, as opposite to search in the whole route. On the other hand, the AOIs not inferred in the T12 cannot be detected by analyzing the tracks speed.

From Figure 4, we can say that the precisions are over 0.8 in three traces and from 0.5 to 0.7 in other traces. We found there were several areas which were inferred as AOIs by the system but were not provided by the traveler. We now discuss the reasons. In the traces T3, T7, T8 and T11, the false inference is caused because the system inferred the area near the traveler's home as an AOI. This kind of erroneous inferences can be avoided by using the user context, i.e. the user's home location.

We consider that recalls are more important than precisions in the AOI inference. When an AOI is not inferred, in the worst case the user have to search the area in the whole route. However, when an area with no interest is inferred as AOI, the user can soon realize and just delete the area from the inference result.

6 Conclusions and Future Work

This paper has presented a novel and effective system to infer the AOIs of a travel by analyzing the GPS trace data of the travel. The system provides a fast access to the inferred AOIs, which helps users in the tasks of examining and modifying travel records. The results show that the system is feasible and provides a helpful inference. The results also show that by using the GPS trace data the user cannot receive a perfect inference because whether an area has interest or not depends on the user subjective opinion. In order to improve the inference, some other parameters can be used, such as the user context (user's home, user's work,…).

As a future work, we are currently developing an inference method based on the density of the points in an area, as opposite to the current method which is based on the density of the points in a fragment of the trace.

Acknowledgments. In the development of the prototype system, we were supported by Yahoo Japan Corporation. We would like to thank Yahoo Japan Corporation. This work was also supported by JSPS KAKENHI 20509003.

References

1. Takahashi, N.: An Elastic Map System with Cognitive Map-based Operations. In: Peterson, M.P. (ed.) International Perspectives on Maps and Internet. Lecture Notes in Geoinformation and Cartography, pp. 73–87. Springer, Heidelberg (2008)
2. Yamamoto, D., Ozeki, S., Takahashi, N.: Focus+Glue+Context: An Improved Fisheye Approach for Web Map Services. In: Proceedings of the 17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, Seattle, Washington, pp. 101–110 (2009)
3. Martinez Lerin, P., Ozeki, S., Yamamoto, D., Takahashi, N.: A Travel Route Editor on a Focus+Glue+Context map. In: 1st International Workshop on Pervasive Web Mapping, Geoprocessing and Services WebMGS (2010)
4. Schlich, R., Axhausen, K.W.: Habitual travel behaviour: evidence from a six-week travel diary. Transportation 30, 13–36 (2003)
5. Zhu, T., Wang, J., Lv, W.: Outlier mining based Automatic Incident Detection on urban arterial road. In: Mobility Conference (2009)
6. Wang, H., Zou, H., Yue, Y., Li, Q.: Visualizing hot spot analysis result based on mashup. GIS-LBSN, 45-48 (2009)
7. Wolf, J., Guensler, R., Bachmann, W.: Elimination of the Travel Diary: An Experiment to Derive Trip Purpose from GPS Travel Data. Transportation Research Record 1768, 125–134 (2001)
8. Ashbrook, D., Starner, T.: Using GPS to learn significant locations and predict movement across multiple users. Personal and Ubiquitous Computing 7(5), 275–286 (2003)
9. KML, `http://code.google.com/apis/kml/`
10. Google Earth, `http://earth.google.com`

Remote-Evaluation of User Interaction with WebGIS

Jens Ingensand and François Golay

GIS-Lab, Swiss Federal Institute of Technology (EPFL), Station 18
1015 Lausanne, Switzerland
{jens.ingensand,francois.golay}epfl.ch

Abstract. In this paper we propose a framework that characterizes user interaction with webGIS. Furthermore we present a technique to remotely test and evaluate such systems. Our framework does not only include parameters describing the usability of a system, but refers to factors that we argue influence interaction and satisfaction substantially, such as the user's background and previous experience, or the user's computer equipment. We have developed tools and techniques to test webGIS remotely so that users can evaluate the system in their own environment rather than an intrusive laboratory. This remote-testing environment allows us to record parameters such as task completion time, the rate of interactions but also the type of operating system and network connection used during the evaluation. In a case study we captured the interaction of more than 300 users with a system whose purpose was to locate points of interest in a region of Switzerland. The data that was gathered during this remote-evaluation was used to validate and verify the conceptual framework. We have found out that parameters such as the user's experience with similar systems or the type of input device and connection speed significantly influence both the interaction and the user's satisfaction. The tools and the framework that we present in this paper are useful for developers and designers of webGIS, but also for researchers who are interested in evaluating the usability of such systems.

Keywords: WebGIS, User Study, Usability, Remote Evaluation.

1 Introduction

In the beginning of the 1990'ies Human-Computer Interaction (HCI) was a well-discussed subject in the community of geospatial sciences. Researchers such as Traynor and Williams [1] have analyzed and criticized how GIS' interfaces are designed. However at that time, HCI-related issues such as usability or spatial cognition were mostly addressed with theoretical concepts rather than with concrete user-studies. Since that time the user interface of most major desktop GIS has not changed.

In recent years the conception and development of webGIS for different purposes has become a widespread trend. The development went from simple mapping applications, offering basic navigation tools to complex collaborative systems where several users can add and modify information. [2]

Initially the design and the functionality of webGIS were highly inspired by desktop GIS in terms of interface design and functionality. Yet webGIS are used in an Internet-context and the design and functionality (e.g. hyperlinks) of standard web

K. Tanaka, P. Fröhlich, and K.-S. Kim (Eds.): W2GIS 2011, LNCS 6574, pp. 188–202, 2011.

pages is usually quite different to the design and functionality of a desktop GIS. This fact resulted in a variety of different interface-types and interaction manners. [3]

The usability of webGIS is crucial for its success since such applications are often intended to be used by people who are not necessarily experts in GIS. Several researchers [4,5,6] have found out that there are still many usability problems, and that there are no clear guidelines and standards for developers on how to design such a system for specific users.

We therefore suggest that it is very important to find out how a webGIS' interface should work and be designed by analyzing how users utilize different systems. All studies that are known to the authors of this paper did not focus on the question who exactly the users are in terms of their personal experience, skills, background, age, gender and so forth. Unwin [7] for example states that the community knows very little about the user. In HCI this point "to know the user" has already been stressed by several authors (e.g. Preece [8]).

The consideration of a webGIS as a system that is composed of the users computer and the system itself is another fact for which we think not enough attention has been paid to. We suggest that factors, as for instance the input device (a touchscreen or a wheel mouse) influence user performance, interaction behavior and satisfaction.

The approach we present in this paper focuses on user characteristics, the context of use and the influence these matters have on usability and interaction. Usability itself aims at increasing aspects of a system such as a system's effectiveness, efficiency and user satisfaction [9]. Other definitions of usability include the parameters speed of performance, time to learn, retention over time, rate of errors by users and subjective satisfaction [10], or efficiency learnability, memorability, errors/safety and user satisfaction [11].

A specific systems' usability can thus be measured and described by the parameters that are given by the definition. In our approach we want to go a step further and not only consider a webGIS with the goal of improving its usability in mind by detecting usability problems, but also describe how interaction and satisfaction are influenced by user-specific factors such as the user's experience with similar systems.

In order to analyze the different aspects, we put up a system of parameters which considers the user, his context, his computer system, the interaction itself, the user's performance and behavior and also the satisfaction.

One key point in our approach is that we do not test a specific system in lab conditions, but we let the users test the system with their own computer equipment. In order to do this we have developed several tools that allow us to record user interaction remotely.

This paper is structured as follows: in the next paragraph we will discuss how other researchers have addressed the subject. We will then present a user-study where we have developed different tools to detect the user's interaction with a specific system. In the next paragraph the gathered data is analyzed and the results are presented. Finally we conclude with perspectives and ideas for future work.

2 Related Work

There are several researchers who have analyzed user interaction in different contexts. However, relatively few evaluations analyzing usability and interaction aspects have been made within the field of webGIS.

Wachowicz et al [4] measured the usability aspects, user satisfaction and user performance (the speed of performance, the rate of interactions and the rate of error) in a case study. The systems evaluated were the systems mappy and map24. There were eight participants who performed the usability test.

The authors came to the conclusions that users, after having used one of the two systems, tried to use the same strategy while interacting with the second system in order to solve the same task. According to the authors this suggests that users generate familiarity during the use of a system and that this fact could give both positive and negative effects.

Another evaluation of similar systems (Google Maps, MSN Maps, Mapquest and Multimap) was conducted by Nivala et al [5]. The authors focused on finding usability problems and on proposing guidelines for the user interface design. The study was conducted with 24 test users (eight 'general' users, eight experts in cartography and eight usability experts) and more than 400 usability problems were detected. Furthermore, the authors stressed the importance of familiarity with such applications.

A different approach was presented by You et al. [3]. The authors analyzed the most essential navigation functions of a webGIS: zoom and pan. You et al. found out that there are different manners these functions are implemented and used in today's systems. For the zoom functions these are:

- zoom by re-center (the map zooms to the point that has been clicked on the map) and original center (the map zooms to the center of the map)
- zoom-in by marquee (the map zooms to a rectangular area that has been selected)
- zoom by fixed scales (the user has the choice of some fixed scales)

For the pan functions there are four possibilities:

- pan-buttons that are grouped around one location
- pan-buttons that are distributed around the map
- clicking on the map re-centers (pans) the map
- moving the map by dragging it pans the map

You et al [3] put up an inventory of ten different systems in order to find out how pan and zoom functions are implemented. The comparison showed that in all cases at least two different zoom functions were available. In nine of ten systems at least two different pan-functions were implemented.

In order to compare the different functions' efficiency, the authors implemented four different systems showing the same (fictive) map. The four systems were capable of recording the user's interactions. The most important results of the study were:

- the original center zoom is more efficient than the re-center zoom (however there was no significant difference in terms of satisfaction)
- the participants found that the zoom icon (a magnifier) fitted the original center zoom better than the re-center zoom
- inexperienced users may think that they are to move the map with the pan-buttons, but not the frame; distributed pan buttons might help the user to move in the right direction; however the distance that has to be done with the mouse is longer than with grouped pan-buttons.

All three experiments (Wachowicz et al [4] You et al [3] and Nivala et al. [5]) have in common is that they have been run in lab-conditions. User interactions were captured either by a camera [5], a specific desktop-recording software [4] or by the system itself [3].

The question if remote testing of web sites could yield similar results as testing within lab-conditions was discussed by Tullis et al. [12]. The authors found out that the behavior of test users within lab conditions is very similar to that of remote users and that the potential larger number of users in remote testing can produce more reliable results. An issue that was discussed by Tullis et al. [12] was that remote testing in many cases requires the user to download a specific web-browser that is capable of recording the user's clicks and the user's interface. In order to facilitate remote testing the authors at that time decided to only use standard html and javascript and were thus limited in the yielded data (Javascript at that time was more limited and less supported by web browsers than today) However the advantage of remote testing is clearly that a higher number of users is more easy to reach and that users are not disturbed by the lab environment.

3 Theoretical Framework

We argue that the interaction of users with webGIS is a complex field of study. Usability problems for instance may have several causes that are not only related to the design of the interface in terms of the functions or the graphical design, but also to an earlier experience with similar systems.

In an earlier case study we had found out that the way users interact with a system can vary from user to user since there are almost always several ways of interacting and solving tasks. [13] Moreover we had found out that the interaction of one user with a system can vary over time and that users can change behavior and strategies during the interaction.

The framework that we propose in this paper therefore takes into account who the users are in terms of age, gender, computer skills, but also in terms of experience with geospatial information and systems. Another very web-application-specific, but neglected fact is that users do not have standardized computer systems used to access these systems. Computers vary and the interaction with a webGIS on a touch-screen is different to the interaction with a wheel-mouse. In particular webGIS are highly subjected to computer performance since such systems, as opposed to "standard" html web pages are more intense in bandwidth consumption (e.g. due to voluminous raster-data that needs to be transferred) and computing power (e.g. for map-rendering). Thus our framework basically consists of four fields of parameters:

Parameters 1: The user

- Demographical parameters (e.g. age, gender, handedness)
- The user's experience and skills regarding geospatial information and systems

Parameters 2: The system

As a system we consider both, the webGIS with aspects such as graphical design and functions, but also the user's computer with features such as bandwidth or input devices (mice, touchscreen, etc.)

Parameters 3: The interaction with the system

- User performance (e.g. task-completion time, errors)
- User behavior (e.g. interaction strategies)

Parameters 4: The user's satisfaction

User satisfaction can be subdivided into parameters such as the overall satisfaction with the system, but also with specific features such as the graphical design or functions.

We argue that the analysis of the different parameter's interconnectedness is an important process that helps understanding how users interact with webGIS. Furthermore the knowledge of a parameter's importance and its connections to other parameters helps weighting the results of a usability study and assists the developers of future systems to put their focus on specific aspects of a system.

4 User Study

In order to analyze the importance and the interconnectedness of the different parameters that we have identified above, we have conducted an experiment that enabled us to monitor, collect analyze most of these parameters. The subject of this user-experiment was a system that we had developed to easily point out and save locations in Switzerland.

Zooming in and out was possible through
- zoom buttons. (a single click zooms out/in, keeping the original center)
- the mouse wheel (moving the mouse wheel up or down zoomed in or out)
- fixed scales (nine choices indicated with blue bars; the actual scale was highlighted by a red color)

Panning was implemented as follows:
- a single click on the map pans it (setting the location as a new center)
- arrows for panning were distributed around the map

Fig. 1. A screenshot of the system's interface

The function of pointing out locations was implemented with the metaphor to "drop" a pin on the map. A button named "drop pin" invited the user to click on it. Once the user had clicked on the button, the mouse cursor transformed into a pin. After the user had clicked on the map, the pin stayed on the map and the mouse cursor changed back again. At the same time a new button appeared, requesting the user to confirm the location.

All elements (buttons, map, etc.) had tool-tips associated, which appeared after 1-2 seconds, indicating what the user could do with it. Fig 1 shows a screenshot of the system's interface.

5 Analytical Methods

Before the system was included in the questionnaire system, a major goal was to evaluate it with real-world users. This testing-opportunity offered the possibility to not only test the system for its usability, but also to record and analyze the parameters described in our conceptual framework.
For the recording of the user interactions two tools were implemented:

- A tool that recorded the server's activity (the server that hosts the system)
- A tool that recorded the interaction with the client (the interface of the system loaded a web-browser)

The first tool was based on the log-file generated by the Apache web-server. In effect, the log-file contains all requests that the server receives along with a time stamp. Fig 2 shows an example of such a line.

```
[1] 70.52.205.158 - - [2][31/May/2006:15:53:46
+0200] "GET [3] /mobility.php?
bbox=561553.0320588425,137880.742787655;56155
3.0320588425,137880.742787655&tool=zoomin&laye
rs=cn,ortho,usr_sel&_= HTTP/1.1" 200 20044
```

Fig. 2. A sample line in the Apache-log file; important parameters:

[1] The IP-address: Who tried to access the web-server
[2] The exact time when the access occurred
[3] The URL that the client requested (=> what the user was doing & seeing)

These queries can be parsed into a more human-readable format and displayed as interaction-protocols. (Fig. 3.) Our tool also generates statistics of how many times a tool has been used or how much time the interaction took, etc.

The IP-address was used to find out the approximate Internet connection speed the user had at the time of the evaluation and also the approximate location of the user. We used IP to location databases (available freely on the Internet) and the host name to find out this data. Users with fixed EPFL IP-addresses were classified as the users with the highest bandwidth; users with a Swiss ADSL or DSL connection were classified as users with a medium bandwidth. All other users (e.g. foreign Internet connection) were classified into a third category.

17:19:13-2008-09-05	0	ex1	1:47200	1968	init
17:19:16-2008-09-05	3	ex1	1:94500	1968	zoom out (mousewheel)
17:19:24-2008-09-05	11	ex1	1:47200	1968	zoom in
17:19:26-2008-09-05	13	ex1	1:23600	1968	zoom in
17:19:27-2008-09-05	14	ex1	1:11800	1968	zoom in
17:19:30-2008-09-05	17	ex1	1:23600	1968	zoom out (mousewheel)
17:19:36-2008-09-05	23	ex1	1:11800	1968	zoom in
17:19:40-2008-09-05	27	ex1	1:11800	2013	RECENTREgszQueryCoords=309,191
17:19:50-2008-09-05	37	ex1	1:11800	2013	QUERYgszQueryCoords=74,177
17:19:53-2008-09-05	40	ex1	1:11800	1418	RECENTREgszQueryCoords=143,166
17:19:54-2008-09-05	41	ex1	1:23600	1418	zoom out
17:19:57-2008-09-05	44	ex1	1:11800	1418	zoom in
17:20:18-2008-09-05	65	ex2	1:47200	29344	init
17:20:21-2008-09-05	68	ex2	1:47200	29060	RECENTREgszQueryCoords=160,181
17:20:23-2008-09-05	70	ex2	1:94500	29060	zoom out
17:20:26-2008-09-05	73	ex2	1:189000	29060	zoom out
17:20:29-2008-09-05	76	ex2	1:378000	29060	zoom out
17:20:36-2008-09-05	83	ex2	1:378000	251	RECENTREgszQueryCoords=349,11
17:20:41-2008-09-05	88	ex2	1:378000	604	RECENTREgszQueryCoords=299,231
17:20:42-2008-09-05	89	ex2	1:189000	604	zoom in (mousewheel)
17:20:44-2008-09-05	91	ex2	1:94500	604	zoom in (mousewheel)
17:20:46-2008-09-05	93	ex2	1:47200	604	zoom in (mousewheel)
17:20:48-2008-09-05	95	ex2	1:23600	604	zoom in (mousewheel)
17:20:51-2008-09-05	98	ex2	1:11800	604	zoom in (mousewheel)
17:20:53-2008-09-05	100	ex2	1:5900	604	zoom in (mousewheel)
17:21:03-2008-09-05	110	ex2	1:3000	604	zoom in (mousewheel)

Fig. 3. An interaction protocol for one session

The tool that records client-activity is a Javascript-library that is loaded when the user accesses the system. The tool samples the user's cursor position and the pointing device's interactions (e.g. clicks) about 40 times a second. The generated data is consecutively sent to a second server.

After the interaction, the data from the database was used to generate parameters such as the cursor speed, the cursor path length or the time spent on specific elements of the interface (e.g. the percentage of time spent on the map itself and the one spent on the tools around)

Fig. 4. The cursor path of a user solving one task (higher speed = red color, lower speed = blue color)

With these two tools it was possible to re-generate the user's screen at any moment of the interaction since the cursor position and clicks were recorded, and the map that the user was seeing could be re-created (all map-generating queries were recorded in the log-file). A sample of the data is given in Fig 4.

6 Evaluation

The evaluation of the system was conducted as a remote online-evaluation. A mail was sent out to about 800 people using mailing-lists (EPFL staff and students). 331 users finally took the time to do the evaluation and to complete the questionnaires.

Table 1. Number of users according to age and gender

	< 20 y.o.	21 – 30 y.o.	31 – 40 y.o.	41 – 50 y.o.	51 – 60 y.o.	*Total*
Male	82	140	5	0	1	*228*
Female	42	57	2	0	2	*103*
Total	*124*	*197*	*7*	*0*	*3*	*331*

Out of these 331 users 228 were male and 103 female. 124 users were younger than 20 years old, 197 users between 21 and 30, 7 users between 31 and 40 and 3 users between 51 and 60 years old (see table 1)

The users had the choice to do the evaluation in English, French or German. At first the users had to complete a questionnaire with demographical questions (age, gender, etc) and questions about their computer skills and about their experience with similar systems.

After the questionnaire the users were required to point out two locations on the systems' map. The original map was centered on the city of Lausanne. The first location to point out was the EPFL (about 5 km outside Lausanne) and the second location was the train station in the city of Yverdon (about 40 km north of Lausanne). During these two tasks the two tools mentioned above recorded the user's interactions.

Thereafter the users had to complete a second questionnaire with questions about their satisfaction concerning specific elements (e.g. the graphical design or the functionality). Users were required to rate different aspects according to their satisfaction. During each evaluation about 80 parameters out of the four categories of parameters mentioned in our framework were measured. The most important parameters are:

Parameters 1: The user
For a user characterization we used the parameters age (in 5 groups), gender, language (French, English or German), left-or right-handedness, estimated level of computer skills and experience with similar systems (we calculated an index that relies on the choice if users had used each of the 9 given alternatives of similar systems: never (=0 points), sometimes (=1 point) and often (=2 points)).

Parameters 2: The system
The user's computer equipment was characterized by the following parameters: the type of the user's pointing device (e.g. wheel mouse, touchpad), the browser type, the type of operating system and a rough measure for the bandwidth.

Parameters 3: The interaction with the system
As a characterization for the user's performance we computed the time it took to complete each task. Moreover we considered locations that we wrongly pointed out by the user as errors. The parameters frequency of clicks on each of the tools and on the map, the time that was spent with the mouse cursor on the map or on the tools around, cursor speed and cursor path length were taken as measures for user behavior and strategies.

Parameters 4: The user's satisfaction
Satisfaction was measured by asking the users how satisfied (on a scale from 1 to 5) they were with the graphical design, the logic design, the design of the tools vs. the functionality that a user expected behind the tools, the map design, the map navigation, the speed and with the system in general.

7 Results

In order to verify and validate our framework we compared the parameters from each group with parameters from another group. For the comparison we used the Wilcoxon rank sum test. The choice fell on this test since it is robust for comparing non-normally distributed samples. The null hypothesis to be tested is that two populations are identical with respect to their medians. A p-value smaller than 0.05 rejects the null hypothesis at a significance level of 5% [16] meaning that the difference between the groups is significant.

Parameters 1 (the user) & parameters 3 (the interaction):
The first question that we addressed in this study is if there is a connection between demographics & user experience with GI & GI technologies (parameters 1 – the user) and user performance & user behavior (parameters 3 – the interaction).

Task completion time (for the two exercises together) was the first parameter that was compared to the user's age, gender and index of experience of similar systems mentioned above. Interestingly the 124 youngest users (younger than 20 years) were the slowest users to complete the exercises. The difference to the group 21-30 years was significant with < 1.00E-6. The 7 users between 31 and 40 years were the fastest. A significant difference was noticed in the speed of performance and the gender of the participants. The 228 male users scored an average of 176 seconds for both exercises; female users (103 users) however had an average of 260 seconds ($p < 1.00E-8$.)

Considering the experience index, we found out that experience with these systems significantly influences the speed of performance. Users with little experience were on average more than one minute slower than users with a good experience (see table 2). The p-values for the comparison between the three groups ranged from $p < 0.004$ to $p < 0.0012$.

Table 2. Previous experience and task completion time

	# of users	Average time per user task1+task2 [s]
little experience (I: 7-11)	112	223.16
fair experience (I: 12 – 16)	194	196.56
good experience (I: 17 – 21)	25	157.12

Regarding errors, we found out that users with a good experience in similar systems (I: 17-21) made on average less errors (0.12 per user) than users with little experience (I: 7-11: 0.26 errors per user). However the p-values for this comparison were only not statistically significant ($p = 0.1$). Moreover we did not find any significant difference between female and male users concerning the amount of errors.

We argue that the rate of interaction (clicks & mouse-wheel interactions counted) is an indirect measure for the user's strategy in spatial navigation. For instance for navigating from A to B a user can choose to simply pan the map until the point is reached (resulting in many interactions) or he can choose to zoom out, pan the map and then zoom in (resulting in fewer interactions). The rate of interaction may furthermore be influenced by the users knowledge of the spatial information displayed on the map. If a user for instance does not know the exact location of point B, he will need to navigate the map until he finds it (resulting in many interactions as well)

In our study the rate of interaction seems to be connected to the experience with similar systems (the comparison of the different groups was significant with p ranging from < 0.026 to $p < 0.029$). Users with little experience (experience index between 7 and 11) needed 30.3 interactions on average to solve both tasks while users with fair experience (experience index between 12 and 16) needed 29.6 interactions and users with a good experience (experience index: 17-21) only 22.4 interactions. Due to the fact that users with a good experience were also faster than the other users, we conclude that the strategy that experienced users have induces better task completion time.

When we investigated the time the users spent with the mouse-cursor on the map itself and the time the users spent outside the map (on tools, buttons, etc) we found out that women spent 73% of the interaction time on the map while men spent 79% of the interaction time on the map ($p < 0.00004$) This result also fits the way female and male users re-centered the map. When considering the proportions of clicks on the arrows around the map to the clicks on the map, we found out that for women the rates (clicks on arrows vs. clicks on the map) are 47% to 53%, while they are 35% to 65% for men. ($p < 0.0003$) Women thus tend to use the arrows around the map more than men.

The differences in percentages between the time on the map versus the time off the map were not significant for the parameters experience and age.

Parameters 2 (the system) & parameters 3 (the interaction):
After analyzing user performance and strategies we wanted to find out if parameters 2 (the system as a whole) had a possible influence on user performance & user behavior

(parameters 3). At first we focused on the users' pointing devices. Most users (198) utilized a wheel mouse for the experiment. Touchpad users (91 users) were the second largest group and normal mouse (without a wheel) users were with 34 users the third largest group. Other input devices such as a mouse stick (4 users), a touch screen (2 users), a trackball and a Playstation-input device (1 user each) were used.

When we compared the task completion time to these groups, we found out that there is no significant difference between the users of wheel mice and normal mice, but between mice users (both mouse types) and touchpad users ($p < 0.04$ for both mouse types vs. touchpad) Touchpad users were on average 36 seconds slower than for instance wheel mouse users. This suggests that the wheel mouse is a more efficient device for the system than the touchpad. Considering the rate of errors, the type of input device did not have any significant influence.

The user's Internet connection speed was the next parameter that was analyzed. We had categorized the users into three groups: 65 users connected to the same network as the system's server at EPFL (these users had thus the fastest connection speed), 232 users connected to the internet through a Swiss ADSL or DSL connection and 34 users in foreign countries.

We assumed that users in foreign countries would have a slower connection speed since there was at least one user in a foreign country who mentioned in a comment that his connection speed was slow. However we did not find any significant difference in task completion time for users who were using the system from a foreign country and users with a Swiss ADSL or DSL connection ($p = 0.074$). Yet there were significant differences in task completion time between users with an EPFL connection and the other two groups ($p < 0.0002$ for Swiss ADSL or DSL users and $p < 0.00003$ for foreign hosts).

We can thus conclude that the Internet connection speed in our case has an influence on the task completion time due to the fact that the system is reacting faster with a fast connection speed.

The pointing device type did not have a significant influence on the time the users spent on the map itself with their pointing device cursors, and the time the users spent outside the map. Neither the rate of interaction (clicking and mouse-wheel interactions) seems to be connected to the type of the pointing device. Interestingly Linux users spent on average 84% of the interaction time with their cursors on the map; Windows users only 79%. This fact however can be dependent on the fact that 17 out of 20 Linux users were men (and men spent more time on the map than women).

Parameters 1 (the user) & parameters 4 (user satisfaction):

We asked the users to rate the graphical design, the logic design, the design of the tools vs. the functionality that a user expected behind the tools, the map design, the map navigation, the speed and the general satisfaction. Grades could be given from 1 (very dissatisfied) to 5 (very satisfied)

All across the different features, most users rated between the options "satisfied" and "neither satisfied nor dissatisfied". However an interesting finding was that the three users between 51 and 60 years of age were more satisfied than all other groups. This fact was found throughout all specific features such as system speed, map navigation, etc. but also with the system in general (average rating: 4.3).

Experience with similar systems did not have any significant influence on the user's general satisfaction with the system, but on the user's satisfaction with map navigation. (Fig. 5). Users with fair experience were less satisfied on average with map navigation than users with little or good experience.

Another noteworthy result is that users who used the system in English (23 users) were less satisfied with the system in general than German language - (26 users) and French language users (Fig. 6). This difference was also visible for all the other specific satisfaction ratings (design, map navigation, etc.).

There was no notable difference in satisfaction for the parameters gender, left-right handedness and the self-estimated level of computer-skills.

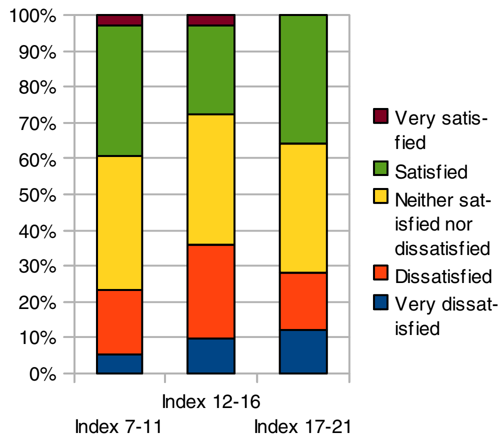


Fig. 5. Percentage of the user's satisfaction with map navigation according to different levels of experience with similar systems

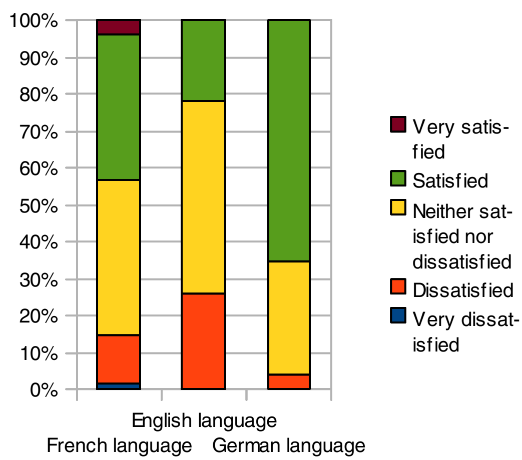


Fig. 6. Percentage of the user's satisfaction with the system in general according to the language they used

Parameters 2 (the system) & parameters 4 (user satisfaction):
An issue for dissatisfaction was map navigation. The average rating of all users for this feature was 2.95. Compared to other subjects such as the system speed (3.5), the design of buttons and icons (3.6) or even the overall satisfaction rating (3.3) map navigation seemed to generate problems. A closer look at the comments showed that 30 out of 41 comments were made about the manner re-centering was implemented in the system. These 30 users preferred to re-center the map using a tool that can be used to drag the map. Some proposed an icon in the shape of a hand for this tool.

Interestingly, the type of the user's pointing device had an influence on the satisfaction regarding map navigation. Touchpad users gave an average rating of 3.09 while normal mouse users (2.94) and wheel mouse (2.89) users gave a worse average rating. The difference between mouse users (wheel mouse + normal mouse users) and touchpad users was significant ($p < 1.00E-12$). This result suggests that mouse users seem to prefer dragging the map while touchpad-users seem less disturbed by clicking on the map. We argue that this fact is highly related to the fact that dragging on a touchpad appears to be physically more demanding than with a mouse.

The speed of the users' Internet connection was a parameter that had a significant influence on the user's satisfaction regarding the system's speed. The differences between all three groups (fast EPFL connection, slower Swiss ADSL or DSL connection and foreign connection) were significant with p-values between 0.0005 and 0.0146. Users with the fastest connection speed gave the average grade 3.8 while users with a foreign connection gave the average grade 2.9. However the fact that users with a foreign Internet connection were less satisfied with the system's speed is probably not only related to their connection speed, since a foreign connection not necessarily is slow and the task completion time was not significantly different between users with a Swiss ADSL or DSL Internet connection and a foreign internet connection. Yet some users who navigated the system with a foreign Internet connection mentioned in a comment that they were unsatisfied with the system's speed due to their Internet connection speed.

8 Conclusions

In our study we were able to detect significant differences between different users regarding their interaction performance and behavior, but also regarding their satisfaction. We therefore suggest that the results are not only valid for the interaction with the webGIS we have described above, but also for other similar systems. However, the user study was conducted with participants who were all within an academic context. We think that this fact might have influenced parameters such as computer skills or experience with similar systems since the particular context involves a frequent use of computer equipment.

The evidence we have found in our data confirms that it is crucial to know the user for the conception, design and implementation of webGIS. The parameters gender, age, experience with similar systems, type of user's pointing device and Internet connection speed do have an influence on interaction and satisfaction. Our evaluation thereby validates the relevance of the framework that we have established.

Moreover we argue that the user himself does not reflect over the system as a composition of different components (interface, input device, Internet connection etc) but perceives the system as one entity. We therefore suggest that a webGIS should not only be developed with the system's web-interface in mind but also with the consideration of the user's computer equipment. We argue that the data we collected still offers more possibilities for analysis and interpretation. For instance, we did not make a difference between the two tasks that the users had to do, but we considered errors and task completion time for both tasks together. In the future, we therefore aim at analyzing the possible different strategies for the two tasks and the possible differences between users.

Furthermore we also want to focus on the connection between spatial cognition and navigation. Several researchers such as Cooke [17] and Chen et al. [18] have discussed the correlation between eye movement and the mouse cursor. In Cookes' study mouse movements matched 69% of eye movements. Chen et al. found out that in 75% of chances the eye will follow the mouse.

Applied to geospatial information systems, Mac Aoidh and Bertolotto [19] used mouse interactions in order to detect a user's interest in spatial features. McArdle et al. [20] even went a step further and developed a system that takes both the recorded user interaction (i.e. mouse movements and mouse clicks) and the user's context (e.g. in terms of their location and device) into account in order to generate a reliable user profile. Such a user profile can thereafter be useful for the dynamic personalization and adaptation of the content of a map. However it should be mentioned that the usage of remote recording tools also raises privacy concerns: in our evaluation, users agreed upon the statement that the interaction would be recorded, however the recording of user interactions without the user's knowledge could represent a considerable infraction on his privacy.

The vast majority of our users used a mouse to interact with a rather simple map-navigation system, however in recent years touch screen input devices have become increasingly popular. A suggestion for future studies is therefore to focus on this kind of devices in order to find out how finger movements (e.g. single or multi-touch) are related to the cognition of space. Further interesting parameters for investigation are more complex tasks such as searching for information or the influence of the resolution and the size of the display.

References

1. Traynor, C., Williams, M.G.: Why Are Geographic Information Systems Hard to Use? In: ACM/SIGCHI Mosaic of Creativity (CHI 1995), Denver, Colorado, USA (1995)
2. Neumann, A.: Web Mapping and Web Cartography 2008. In: Shekhar, S., Xiong, H. (eds.) Encyclopedia of GIS, pp. 1261–1269. Springer, Heidelberg (2008)
3. You, M., Chen, C.W., Liu, H., Lin, H.: A usability evaluation of web map zoom and pan functions. International Journal of Design 1(1), 15–25 (2007)
4. Wachowicz, M., Cui, L., Vullings, W., Bulens, J.: The Effects of Web Mapping Applications on User Satisfaction: an Empirical Study. In: Peterson, M. (ed.) International Perspectives on Maps and the Internet, pp. 397–412. Springer, Heidelberg (2008)
5. Nivala, A., Brewster, S., Sarjakoski, L.T.: Usability Evaluation of Web Mapping Sites. The Cartographic Journal 2, 129–138 (2008)

6. Skarlatidou, A., Haklay, M.: Public Web Mapping: Preliminary Usability Evaluation. GIS Research, Nottingham (2005)
7. Unwin, D.J.: Fiddling on a different planet? Geoforum 36, 681–684 (2005)
8. Preece, J., Rogers, Y., Sharp, H.: Interaction Design - Beyond Human- Computer Interaction. John Wiley & Sons, Chichester (2002)
9. ISO 9241, Ergonomic requirements for office work with visual display terminals (VDT's). Part 11. Guidance on usability (1994)
10. Shneiderman, B.: Designing the User Interface. Addison-Wesley Publishing Company, USA (1998)
11. Nielsen, J.: Usability Engineering. Academic Press, London (1993)
12. Tullis, T., Fleischmann, S., McNulty, M., Cianchette, C., Bergel, M.: An Empirical Comparison of Lab and Remote. In: Usability Testing Usability Professionals Association Conference, Orlando, July 8-12 (2002)
13. Ingensand, J., Golay, F.: User Performance in Interaction with Web-GIS: A Semi-Automated Methodology Using Log-Files and Streaming-Tools. In: Bernard, L., Christensen, A.F., Pundt, H. (eds.) The European Information Society, pp. 433–443. Springer, Heidelberg (2008)
14. Lindgaard, G., Chattratichart, J.: Usability testing: what have we overlooked? In: CHI 2007: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, pp. 1415–1424. ACM, New York (2007)
15. Ericsson, K.A., Simon, H.: Protocol Analysis: Verbal Reports as Data. MIT Press, Boston (1993)
16. Gibbons, J.D.: Nonparametric Statistical Inference. MarcelDekker, New York (1985)
17. Cooke, L.: Is the Mouse a "Poor Man's Eye Tracker"? In: Society for Technical Communication Conference (May 2006)
18. Chen, M.C., Anderson, J.R., Sohn, M.H.: What can a mouse cursor tell us more?: correlation of eye/mouse movements on web browsing. In: CHI 2001: CHI 2001 Extended Abstracts on Human Factors in Computing Systems, pp. 281–282. ACM, New York (2001)
19. Mac Aoidh, E., Bertolotto, M.: Improving Spatial Data Usability By Capturing User Interactions. In: AGILE 2007 (10th AGILE International Conference on Geographic Information Science), Aalborg, Denmark. Lecture Notes in Geo-Information and Cartography, pp. 309–403 (March 2007)
20. McArdle, Ballatore, A., Tahir, A., Bertolotto, M.: An open-source web architecture for adaptive location-based services. In: Proceedings of the 14th International Symposium on Spatail Data Handling, SDH (2010)

Touch2Query Enabled Mobile Devices: A Case Study Using OpenStreetMap and iPhone

Junjun Yin and James D. Carswell

Digital Media Centre, Dublin Institute of Technology, Ireland
yinjunjun@gmail.com, jcarswell@dit.ie

Abstract. This paper describes our mobile spatial interaction (MSI) prototype *Touch2Query* which presents the idea of using the touch screen on mobile devices to assist in performing ad-hoc spatial queries. This approach differs from conventional mobile LBS applications where the query shape (search space) is limited to either a bounding box or radius. Instead, we provide functionality that allows users to interactively draw any desired query shape overlaid on an area of interest directly on a mobile device with their finger by combining vector primitives such as circles, polygons, polylines, and points. With the help of location and orientation aware mobile devices, mobile maps, and real-time distance and area measurements, *Touch2Query* gives the users freedom to perform customised spatial queries on objects/areas of interest while realising a better contextual understanding of their spatial environment at the same time.

Keywords: Touch screen, MSI, OpenStreetMap, Geospatial web-services.

1 Introduction

As the development of mobile technology keeps marching forward, new generation *smartphones* like the Apple iPhone, Google Android, and Nokia Symbian based devices have become increasingly popular (Shek, 2010; Takeuchi and Kennelly, 2010). Position aware micro-sensors (i.e. GPS together with digital compass and accelerometers) are integrated into most of today's state-of-the-art COTS (commercial off-the-shelf) devices together with supplemental locationing technology from Wi-Fi and cell tower signals. As such, these mobile devices are now fully spatially enabled in terms of determining their geo-location with sufficient accuracy and consistency for most outdoor LBS applications (Skyhook, 2010). Since 3G networks became prevalent, fast internet connections have now turned mobile devices into ideal terminals for World Wide Web discovery. Together, these advances bring a variety of new opportunities for development of Location-Based Service applications, which are essentially based on a user's geo-location and delivered as a web-service, for example, "*show me all the coffee shops within 1 km*" or "*what are these buildings around me*".

Most new smartphones are also mobile map ready for navigation purposes, such as Ovi Maps for Nokia phones and Google Maps for Android and iPhones, etc. Using *mashup-maps* is a popular LBS approach, where the results retrieved from the

K. Tanaka, P. Fröhlich, and K.-S. Kim (Eds.): W2GIS 2011, LNCS 6574, pp. 203–218, 2011.

web-service are displayed as annotations on top of the base mobile maps. In this case, the mobile map annotations help users to get to know their *spatial context*; i.e. in relation to the nearby geographic environment surrounding a user's current location, while behind the scenes each request is interpreted as a spatial query (search) to retrieve corresponding geometries and other associated information. The results from the spatial queries are determined by the spatial relationship between the source query shape/window and the spatial data (geometries) repository. Taking the "*show me all the coffee shops within 1 km*" request for instance, it is a type of range query that generates a bounding circle or bounding box with a specified dimension (e.g. radius) as the source query shape. The spatial operator for the query determines whether those geometries (building *footprints* of the coffee shops in this case) have the "within" topographical relationship when compared to the source query shape and subsequently deliver any retrieved results. Currently, there are a couple of well known web-services that deliver such functionality, such as Google Maps and Bing Maps, and applications running on the mobile device built on top of such services, for example, AroundMe (2010) and GeoVector (2010), etc. However, some limitations are still present when performing such types of spatial queries:

1. The results retrieved from such services are entirely dependant on the completeness of the geospatial datasets of the data providers. In some areas, especially smaller cities or rural areas, map coverage is considerably poorer and hence query result accuracies are likewise affected.
2. Even though the bounding query shape has already confined the search space and filtered out those objects that are out of bounds, it assumes that users have at least some local spatial knowledge of their surrounding environment. For instance, if the search radius is too large, in a densely built environment the returned results will be overwhelming, producing display clutter or *information overload*. However, according to Willis et al. (2009), users that use mobile maps for navigation usually have poor local spatial knowledge familiarity. Therefore, if a user is a stranger in a new environment, a 1 km radius query range makes little sense in this context.
3. The query shape is limited to either a bounding box or a circle, and often without the shape's extents actually being displayed to users on the device. In some cases, where users are particularly interested in a certain area, such a query will either include some unnecessary geometries or miss some potential geometries altogether since the users have no freedom to manually adjust the position of the query shape boundaries and move beyond the default rectangle/circle range query parameters.

In relation to the availability and completeness of a geospatial dataset, (Goodchild, 2007) asserts that there is a rising interest in contributing volunteered geographical information (VGI) content to the Web that enriches the amount of accessible spatial data available. One of the more successful VGI examples is the *OpenStreetMap* (OSM, 2010), where the map contents are contributed from volunteers through GPS trajectories and digitized map content, etc. Moreover, OSM is not limited to only street data but also footprints of buildings, point-of-interests (POI), etc. As a result of globally volunteered contributions, the completeness of overall map coverage is growing daily where OSM already shows a significant advantage in this respect over Google Maps and others in many cases (Jacob et al., 2009) and we concur with (Becker and Bizer, 2009) and others that these detailed and free to access spatial datasets offer great potential for developing added-value applications for LBS.

With respect to limitations arising from the restriction of query shapes, a new feature found in some of today's smartphones, i.e. the touch screen, is now available for developing enhanced user interaction functionality on these COTS devices where high precision multi-touch screens can leverage a user's existing experience with human computer interaction (Albinsson and Zhai, 2003; Benko et al., 2006). Even though the screens on mobile devices are smaller than those on a desktop, with high precision selection users can easily perform customized gestures to view map contents, such as pan/zoom/draw etc. Our case in point is the Apple iPhone, which integrates built-in GPS, digital compass, accelerometer, and a multi-touch function enabled touch screen. One of the obvious benefits of using touch screen interaction is that users can now specify a search space by interactively drawing query shapes directly on top of the mobile map display using their finger. This idea mimics using a mouse to draw ad-hoc query shapes for geo-processing in traditional desktop Geographic Information Systems (GIS). On one hand, users can visually see the query shape while interacting with the map contents in real-time. While on the other hand, since it is drawn on top of a mobile map, users get a better awareness of the context of their surrounding environment instead of simply a conceptualized radius.

This paper describes a prototype application, namely *Touch2Query*, which uses the touch screen of an Apple iPhone to perform spatial queries over OSM datasets. It presents the idea using the touch screen as an interface for directly capturing user gestures. Such interaction is translated (interpreted) into source query shapes for further spatial queries in the database. Although at the current stage the geospatial data coverage in OSM is not entirely complete, as VGI moves towards more collaborative input as part of Web 2.0, we see the *collective intelligence* as a promising mobile map and metadata resource for LBS apps. Therefore, this paper also intends to demonstrate the usefulness of OSM data already available for spatial information retrieval in general.

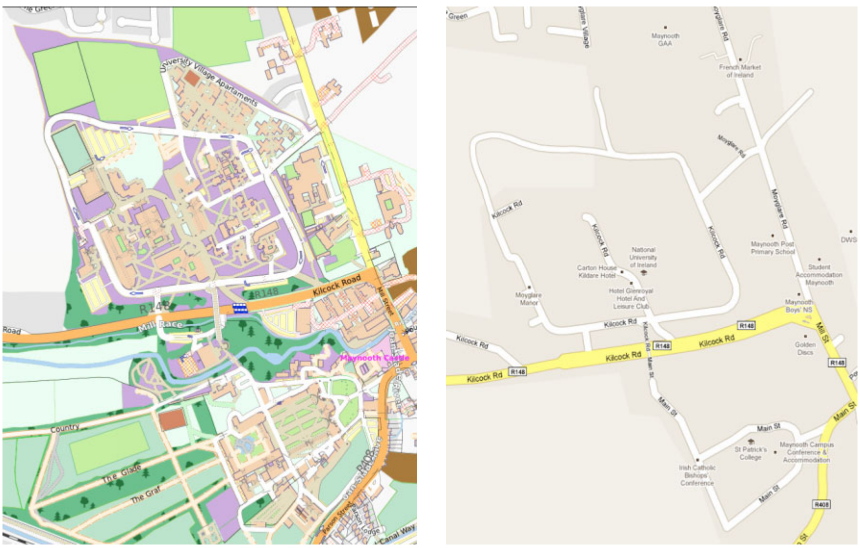
The remainder of the paper is organized as follows: Section 2 focuses on usability and objectives of the *Touch2Query* application from the perspective of geospatial information retrieval on smartphones, where some related applications in this field are introduced. Section 3 gives a comprehensive description of the data integration module, methods and functions that are employed in this application. Detailed demonstration of the implementation on the iPhone is shown in Section 4, and the conclusions and future work is presented in Section 5.

2 Understanding Spatial Information Retrieval for Mobile Devices

2.1 Volunteered Geographical Information (VGI)

The availability of detailed geographic data is critical for delivering comprehensive LBS applications. Most geospatial data, in Europe at least, is collected and controlled by national mapping agencies or private companies, such as Google Maps, Yahoo! Maps, and Bing Maps, etc. However, data coverage over certain areas can still be of considerably poor quality and extent - especially in less populated areas. Some research attempts to overcome this shortcoming looked at imaging and georeferencing public displays of "You Are Here" maps to fill the coverage gap for local navigation purposes (Schöning et al., 2009). But a rapid growth in VGI has also started to fill this gap with OSM being a successful example of this. In relation to VGI, the *citizen-as-sensor* paradigm contributes to OSM by creating, assembling and disseminating geospatial features including streets, highways, buildings, etc. and gradually this collective geospatial information shows a surprising coverage all over the world (Goodchild, 2007; Haklay and Weber, 2008; Haklay and Ellul, 2010). Another vitally important feature of using OSM data is that "*you are free to copy, distribute, transmit and adapt our maps and data, as long as you credit OpenStreetMap and its contributors*" (OSM, 2010), which affords users a lot of commercial freedom when building added value applications on top of it.

Considering this increasing trend of VGI sourced data, this paper aims to demonstrate spatial information retrieval for mobile devices by building upon the base geospatial data supplied from OSM. As a case study, the map data that covers the National University of Ireland, Maynooth (NUIM) and its surrounding areas is targeted. A screenshot of the OSM map coverage is shown in Figure 1(a), while the corresponding map coverage over the same area from Google Maps is shown in Figure 1(b). The detailed OSM data is created by students from NUIM and is available for everyone to use, a clear example of how OSM grows through volunteers contributing datasets (Jacob et al., 2009).



(a) (b)

Fig. 1. (a) Map coverage around NUIM, Ireland from OSM (b) Map coverage for the same area from Google Maps

2.2 Spatial Information Retrieval for Mobile Devices

2.2.1 Query Shapes in Spatial Information Retrieval

By definition, spatial information retrieval involves a query process to determine the spatial relationships between a source query window (shape) and a target data repository, where those objects that satisfy the spatial relationships get returned. For instance, searching for objects (e.g. buildings, streets, etc.) that are within a certain radius from the current location of a user is referred to as a *range query* or *spatial proximity query* in GIS (Mountain and MacFarlane, 2007). In this case, a bounding box or circle with specified radius eliminates those objects that are out of bounds and returns only those that have the "*contains*" spatial relationship when compared to the query shape.

There are already a wide range of web-based services that provide such operations, e.g. Google Maps, Bing Maps, etc., and mobile applications built on top of these services. Taking advantage of powerful search engines, these services and applications can already perform queries on large spatial datasets. However, the query shape in each case is still limited to either a bounding box or a bounding circle with a specified radius. An illustration of these two query types is shown in Figure 2.

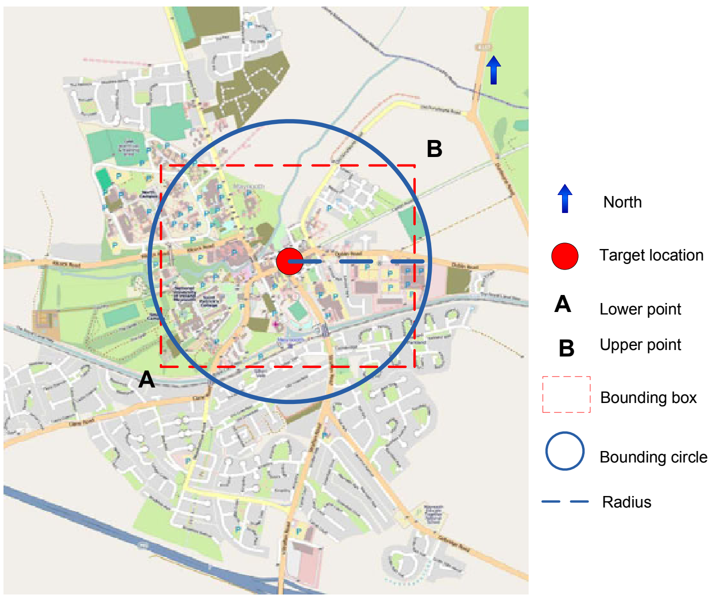


Fig. 2. Using bounding box and circle for range queries

In many cases, both types of query will work to aid users in reducing the information overload problem by filtering information that is out of bounds and thus presumably out of interest. However, if a user is navigating at a large map scale (i.e. zoomed-in to street/building level) in an unfamiliar environment, the specification of radius makes less sense than it would at a smaller map scale (i.e. when zoomed-out). At street level, the users might be particularly interested in nearby areas only where it is hard to know the effect of scale, i.e., the difference between a 100 meter query and 75 meters for instance.

Other innovative applications make use of a combination of integrated GPS and digital compass to perform directional spatial queries where the query shape is a calculated portion of the bounding circle in the pointing direction of the device to simulate a user's actual field-of-view (Gardiner et al., 2009; GeoVector, 2010). A similar approach used in GeoWand (Simon et al., 2007) and 3DQ (Carswell et al., 2010), where returned query results are only the objects visible within the mobile device's pointing direction. Such spatial queries are accomplished by determining whether the pointing vector (a straight line projecting along the pointing direction) intersects with any objects in the database. These approaches aim to involve human interaction when performing the spatial queries and focus on the user's egocentric perspective to improve the query experience. Nevertheless, these approaches rely on a GPS to provide constant and accurate positioning even though any positioning error (of even a few metres) can significantly influence the query results and constant running of the GPS and digital compass has been shown to quickly drain battery life in a mobile device.

We suggest that a true egocentric query experience can never be fully realised if the users themselves cannot fully control the spatial query shape. With fully position-aware mobile devices now available, current location and orientation is a given. Therefore, we conclude that it's now time to provide tools that aid the user to interactively refine their exploration of the environment, instead of "fixing" every query parameter for them. From this point of view, this paper presents the idea of exploiting the touch screen of the mobile device and allow users to draw their own query shapes in their own defined locations (i.e. not limited to their current location) on top of a mobile map. As a result, the customized search space is an arbitrary form. For example, swiping a finger over the mobile map is now considered equivalent to a directional query when pointing the device. The difference is, in this case, the user picks the direction they are interested in querying based on the current map displayed on the device, which may still be positioned and orientated according to the device's sensor inputs, but not necessarily so. This approach has the added advantage that it removes any requirement for these sensors to be absolutely accurate – a common but often false assumption among many existing mobile directional query type applications.

2.2.2 Spatial Database and Web Services

Retrieving comprehensive and meaningful results from spatial queries involves considerably large amounts of data and data processing. Considering the physical limitations of today's mobile device battery power, CPU speed, and data storage space, etc., it is impossible to accomplish all these tasks efficiently on the device itself. Therefore, a client-server architecture is the preferred option where calculation intensive tasks are carried out on the server side, leaving the mobile device as a client to take care of sending requests, receiving responses, and displaying query results to the users (Simon and Fröhlich, 2007; Shek, 2010).

To deal with the large amounts of geospatial data on the server, a spatial Database Management System (DBMS) is required to store, index and manage these data. Most databases today can fulfil this role, such as Oracle Spatial 11g, PostGIS, SQLite Spatial, MySQL, etc. (Zhou et al., 2009). Each of these databases has mechanisms to build spatial indexes on the stored geospatial data which significantly speeds up the query process of discovering spatial relationships (e.g. cover, contain, intersection, etc. between two geometries). In *Touch2Query*, to embrace the possibility that 3D data might be soon available in VGI, we employ Oracle Spatial 11g for this application as it already supports 3D geometries (Ravada and Kothuri, 2009). Although the integrated SQLite database on the iPhone is not spatially enabled (i.e. "geometry" is not a native data type), it can still be used to store shapes of geometries in the form of lists of coordinates and can attach other customized information, such as web links, descriptions, etc. In this case, if datastores are pre-loaded, such an approach can save on Internet connection and other data traffic costs.

Otherwise, a client-server communication architecture is a key element of successful LBS implementations. In this respect, a common method is to use a *request-response* style, which are completed by remote procedure calls (RPC). The *Touch2Query* server delivers Representational State Transfer (RESTful) web-services using TurboGears (tgws 2010), which refers to an architecture where the request is an *http* call and the response can be in different data formats such as XML or JavaScript Object Notation (JSON). A successful example is the Google Ajax search API where the remote call is an http request with specified parameters embedded in the web link and the response data is in JSON format (Google Ajax, 2010). In particular, we utilise a JSON response on the iPhone side simply because we believeve the data format is easier to work with.

We exploit the strategies mentioned above to accommodate customized spatial query shapes. However, in relation to working with OSM data, there are already some useful web-services available. For example, (GeoNames 2010) provides the service *findNearbyStreetsOSM* which finds the nearest street to a given coordinate in the OSM datasets and provides both a JSON and XML response. There are also other interesting services available that the *Touch2Query* prototype utilises to demonstrate the integration of rich and free online resources built on top of OSM data.

3 *Touch2Quey* Application Architecture

The aim of this section is to give a detailed description of the architecture of the *Touch2Query* prototype. An overview of the system is shown in Figure 3. The architecture as a whole consists of three parts: Data integration model, query shape control model, and data exchange model.

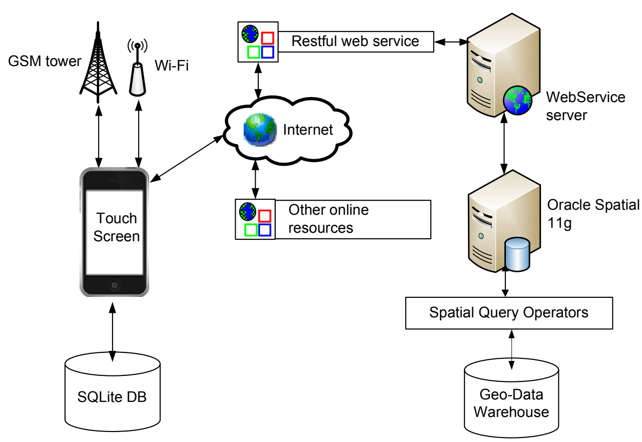


Fig. 3. Architecture of the *Touch2Query* system

3.1.1 Data Integration Model

The spatial data queried in this paper is downloaded directly from OSM, which covers our test area of NUIM and surrounding area, as shown in Figure 1(a). The geometries are in two data types: polylines for highways and streets; and polygons for buildings.

The data integration model consists of three parts: On the server side, both polylines and polygons are stored in the spatial database in two separate tables, where each feature has its corresponding non-spatial attributes (e.g. name, type, etc.) attached. The second part of the data model utilizes the existing online web-services for OSM data, GeoNames in particular. Even though the actual data is stored in the GeoNames server, it is downloaded from the same up-to-date OSM repository. These two cases need Internet connections for data exchange. The third part of the data integration model uses the integrated iPhone SQLite database to locally store detailed information of each building on campus, such as building shape, name, description, etc. There are total 144 building footprints in the NUIM campus; each individual shape is recorded as a collection of coordinates for vertices in the footprint polygon. Therefore, once the building names in the query results get identified, the displayed contents can be derived locally from the SQLite database to save Internet data costs. Another benefit in using SQLite is that a simple range query can be performed using the bounding box; of course in our case the bounding box is user specified using the touch screen. Therefore, a minimum bounding rectangle (MBR) of each building is pre-calculated and the lower and upper points of each MBR are attached as attributes respectively.

3.1.2 Customized Shapes Control Model

For actual spatial query processing, query results are determined according to each spatial operator the user specifies (e.g. cover, intersect, contain, etc.). The spatial operator filters out those objects that do not have the specified spatial relationship with the source query shape and hence aims to reduce the effect of information overload and offer the user a clearer understanding of the surrounding environment.

As mentioned, in *Touch2Query*, query shape specification is not limited to a bounding box or a range radius. Instead, it is an arbitrary shape specified by the user and is drawn interactively via the touch screen. The implemented drawing functions support point, polylines, polygons and circles. To be more specific: a point is when the user touches a specific object (e.g. building) in the mobile map; a polyline is when the user swipes the screen over one or several objects in the map; a circle uses the first touch as the centre and moves to adjust the radius while keeping the finger in contact with the screen; a polygon is drawn when the user touches the screen multiple times in various locations on the mobile map.

The implementation of these query interfaces is relatively straight forward. We attach two layers to the touch screen: one at the bottom is the *map layer* and one at the top is an *empty layer*, which is used to capture any touch gestures from the user. The map layer consists of the iPhone integrated Google Maps. Considering the fact that Google Maps has poor coverage over the test area, we use OSM as our base map. For faster display, a group of geo-referenced OSM tiles are used as overlays on top of the background map. These tiles are split into smaller tiles and stored according to different map zoom levels and behave transparently as part of the native map. The most important role for the top layer is that it interprets the touch point into a latitude/longitude coordinate in the underlying base map. Once the user double-taps the touch screen, the shape is drawn and ready for query processing to begin.

3.1.3 Data Exchange Model

The data exchange model of the system is split into two parts (online and offline) and shown in Figure 4:

1. Web-Services based: The spatial queries over the OSM data from the GeoNames and *Touch2Query* server is accomplished online via RESTful web services, where the function call from the mobile device is a simple http request with specified parameters, such as the coordinates of the query shape; the response data from the server is in JSON format.
2. Local SQLite based: In offline mode, the spatial query using a simple query shape is carried out on the SQLite database. A detailed example query process is described in the following section.

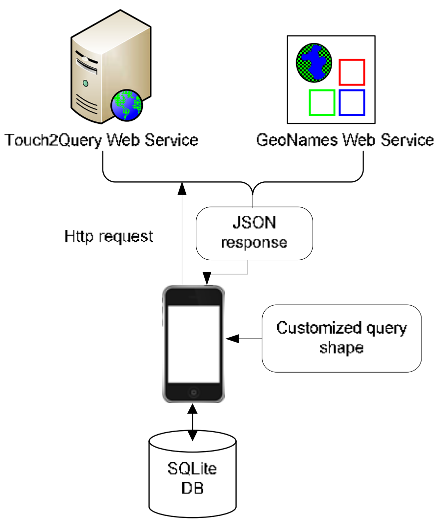


Fig. 4. *Touch2Query* data exchange model

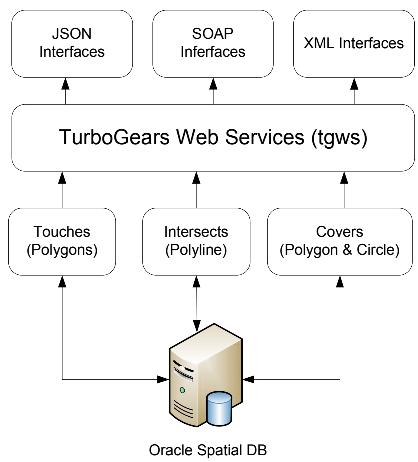


Fig. 5. The *Touch2Query* web-service

In relation to the functions implemented in the *Touch2Query* server, three types of spatial operators are provided, i.e. *"touches", "intersects"* and *"covers"*. To be more specific: In case the query shape is a polygon, the "*touches*" operator treats the polygon as an "outline" instead of a plane and returns those objects (buildings or streets) that intersect with the boundary only. While, "*covers*" treats the shape as a plane and returns those objects that are within the query shape. When the query shape is a circle, it is equivalent to a range query with the radius visible and manually adjustable in our case. When the query shape is specified as polylines, those objects that "*intersect*" with the line will be returned. For instance, a swipe gesture in this case will generate a straight line, which is equivalent to a direction query. The diagram of the *Touch2Query* server shown in Figure 5 is built on top of the *tgws* framework and provides three types of responses (JSON, SOAP and XML) automatically.

As for integrating web-services from GeoNames, the *find nearest intersection OSM* service is adopted where once a point is located in the mobile map, users can use this query to find the nearest street and the next crossing street from the specified location.

As the integrated SQLite database on the iPhone is not spatially enabled, there are no native spatial operators supported. However, with help from pre-calculated MBRs of geometries, simple spatial operators can be managed via SQL selections (Rubin, 2006). In particular, in *Touch2Query*, two types of query shape are supported: point and bounding box (rectangle). The implemented spatial operators determine the topological relationship between the query shape and each MBR of buildings in the database and retrieve the corresponding data.

3.2 Helper Utilities

3.2.1 Basic Geographic Measurement Utility

When performing spatial queries, it is common that users are promoted to specify a search radius, which helps to eliminate the risk of inefficient querying over an entire huge dataset. However, such procedures assume the users have local spatial knowledge familiarity, for instance, understand roughly how far 1 km is on the map. *Touch2Query* avoids this assumption not only by letting users specify the radius by drawing the query shape directly on top of the area of interest, but also providing two geographic measurement tools: measurement for area (polygon) and length (polyline). For example, when specifying a circle or bounding box, the radius or width and height in ground units (e.g. metres) gets updated in real-time respectively.

Fig. 6. Interface for user feedback utility

3.2.2 Utility for User Studies

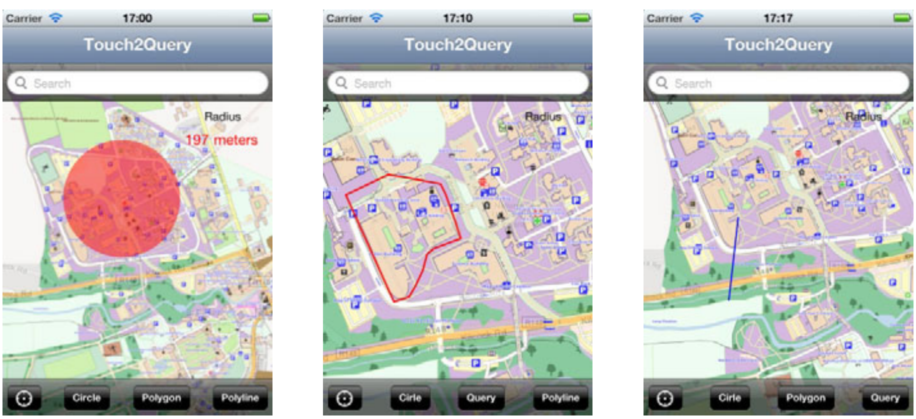
Even though this application aims to give users flexibility while interacting with their spatial environment, query choices are still limited to those that are implemented. Therefore, feedback from users regarding how they like or dislike the functionality, which query functions are preferred, and any additional comments, are important and valuable for future improvements. A *Touch2Query* user feedback utility is therefore embedded in this prototype. Users can grade each function on a scale from 1-5 and submit their comments and suggestions. Such feedback information gets received and stored in the database tied to a unique user id. This allows for further user studies analyzing usage patterns based on the preferences they have graded for future functional improvements. A simple interface for the feedback utility is shown in Figure 6.

4 *Touch2Query* on the iPhone

This section describes the *Touch2Query* prototype as implemented on the iPhone. The integrated Google Map together with the OSM overlay serve as the base maps, the touch screen is used to specify query shapes, the spatial query functions utilize the services from *Touch2Query* server, GeoNames server, and the integrated SQLite database, and user location and orientation awareness is supported by the integrated GPS and digital compass.

4.1 Touch Screen for Customized Spatial Query Shape

The query interface on the iPhone is shown in Figure 7, where the black transparent bar at the bottom contains buttons for different query options. In particular, the left



(a) (b) (c)

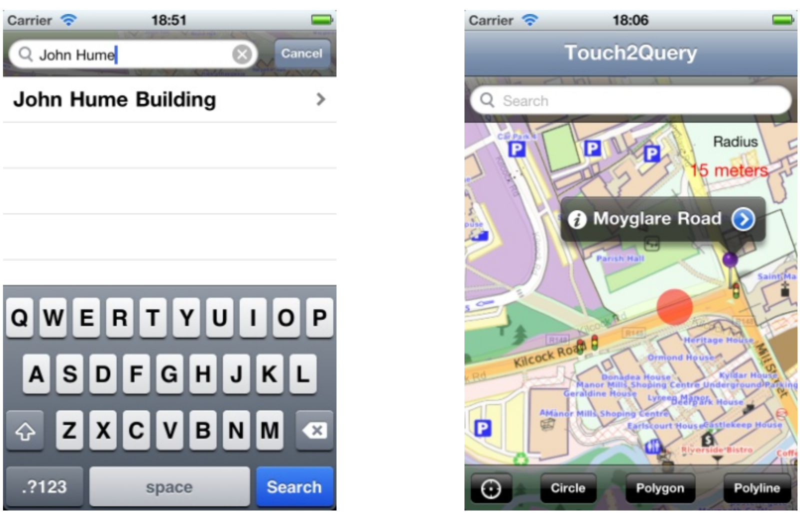
Fig. 7. (a) A query shape using a circle with drawn radius. (b) A query shape in the form of an arbitrary polygon. (c) A straight line drawn using a swipe gesture for the query shape. Note that none of these query shapes need to be necessarily centred on the user's current location and so can be interactively drawn on/over any area of interest.

most button will trigger the device to provide the current location with a single tap onthe button; and the orientation the device is currently pointing in with a double tap. Once the digital compass is working for orientation, the base map will auto-rotate to make the map point in correspondence with the pointing direction of the device.

The customized query shape is then drawn on top of the base map, after the user gets informed of their location and orientation. Any finger touches are captured and interpreted into geo-referenced shapes, i.e. each point is translated into a coordinate on the map. The shape is updated in real-time with the touching gestures. In particular, in Figure 7(a), the shaded radius of the circle is continuously updated in meters to let the users know the exact area they are querying; Figure 7(b) shows a query window as an arbitrarily drawn polygon; In Figure 7(c), the user draws a straight line on the screen to search or buildings and streets that are intersected along the path of the line.

4.2 Data Integration and Exchange Models

The OSM data is downloaded and stored in the spatial database (Oracle Spatial 11g) on the server side and locally on the iPhone using SQLite database respectively. Since the SQLite does not handle geometry as a native data type, the coordinates of the



(a) (b)

Fig. 8. (a) Searchable building information stored in SQLite. (b) The result returned from GeoNames (purple pin) for nearest street intersection from a given point (red dot)

vertices in a geometry is stored as a collection. In addition, the MBR of each building is pre-calculated and the lower and upper point of the rectangle is stored, as well as other information like web links and description, etc. The spatial query process can be accomplished by either sending requests to the *Touch2Query* and GeoNames web-services (Figure 8(b)), or querying from SQLite locally (Figure 8(a)). As a demonstration, an http request sent to GeoNames about the nearest street intersection with the current street location in OSM is shown as:

```
http://ws.geonames.org/findNearestIntersectionOSMJSON?lat=53.382573&l
ng= -6.596088
```

It is a simple URL calling the function name "*findNearestIntersectionOSM*", with specified coordinates for the current street location (red circle) and the response data type in "*JSON*". The returned result (purple pin) is then annotated on top of the mobile map, as shown in Figure 8(b).

4.3 An Example of Execution Flow

This section demonstrates a spatial query using polygons in the *Touch2Query* web-service with a user drawn polygon as a query shape. After specifying the query shape, users are provided with certain query options as shown in Figure 9(a). Taking the "*Intersected Buildings*" option for instance: an http call using the coordinates of the polygon shape is sent to the *Touch2Query* web-service; the server side uses the "*any intersection*" spatial operator to discover any intersected buildings and returns/displays the building names on the iPhone as shown Figure 9(b). Such results can also be visualized as shown in Figure 9(c), where the building footprints are pulled from SQLite (to save on network download time/costs) and drawn in blue.

(a) (b) (c)

Fig. 9. (a) Available options for an arbitrary polygon query shape. (b) Returned results shown in a table. (c) Visualized results displayed on top of the mobile map.

5 Conclusions and Future Work

This paper describes our MSI prototype *Touch2Query* which presents the idea of using the touch screen on today's smartphones to perform ad-hoc spatial queries. The query shape is drawn using finger touches for customized shapes such as; circle, polygons, polylines, and points. With the help of location and orientation aware

COTS devices, mobile maps, and real-time distance and area measurements, *Touch2Query* gives the users freedom to perform spatial queries on their areas/objects of interest with a clear understanding of spatial context (wrt scale) in their environment at the same time. The potential advantages are:

1. While mobile devices do need GPS and digital compass to show current location and orientation, in *Touch2Query* these two sensors do not need to work constantly or accurately as users can explore and search their surrounding area manually, which can also significantly extend battery life on the device.
2. During the process of query shape (search space) specification, users can visually see what the query shape looks like, exactly what sort of area it is covers, etc. and therefore get a more intuitive impression of their space and gain a better egocentric experience.
3. In general, compared to conventional range queries, *Touch2Query* allows more flexibility and freedom for users to discover their spatial environment while reducing information overload at the same time.

In relation to the data integration model used by this application, geospatial data from OSM is employed not only because its coverage of our NUIM test bed is better than others, but also because of its VGI nature in that such data can be freely accessed and distributed in mobile LBS applications. Similarly, the *Touch2Query* server delivers open accessed web-services via a RESTful architecture and offers responses in XML, JSON and SOAP formats. This means with a simple http request, most touch enabled mobile devices including iPhone, can *plug and play*. In addition, this application also integrates OSM web-services from GeoNames to demonstrate the possibilities of incorporating other freely available online web-services on mobile devices.

Future work will consider the following areas:

1. In relation to the employed datasets for this application, the accuracy of the data shall be investigated and validated. It was found that some building and street names are missing while others are associated with wrong names.
2. Although OSM has been evolving fast as part of the VGI phenomenon, at this stage it is still too early to conclude that OSM will replace its commercial counterparts. It is inevitable that we should embrace the trend and make good use of these valuable resources by integrating them into future applications.
3. Although a user feedback utility is embedded in this application, at this moment *Touch2Query* has not been tested by many users. Once more user feedback information is gathered, a detailed user study about usage patterns will be investigated and summarized.
4. Even though this application shows how touch screens can make mobile maps more interactive in 2D, we are living in a 3 dimensional world. Future work will focus on making 3D objects touchable in the mobile device, including being able to draw 3 dimensional query shapes for spatial information retrieval.

References

Albinsson, P., Zhai, S.: High precision touch screen interaction. In: SIGCHI Conference on Human Factors in Computing, pp. 105–112 (2003)

AroundMe (2010), Retrieved from: http://www.tweakersoft.com/mobile/aroundme.html (accessed 2010-09-17)

Benko, H., Wilson, A., Baudisch, P.: Precise selection techniques for multi-touch screens. In: Proceedings of CHI 2006, pp. 1263–1272 (2006)

Becker, C., Bizer, C.: Exploring the Geospatial Semantic Web with DBpedia Mobile. Web Semantics: Science, Services and Agents on the World Wide Web 7(4), 278–286 (2009), doi:10.1016/j.websem.2009.09.004

Carswell, J.D.: 3DQ: Threat Dome Visibility Querying on Mobile Devices. GIM International 24(8), 24 (2010)

Gardiner, K., Yin, J., Carswell, J.D.: EgoViz – A mobile based spatial interaction system. In: Carswell, J.D., Fotheringham, A.S., McArdle, G. (eds.) W2GIS 2009. LNCS, vol. 5886, pp. 135–152. Springer, Heidelberg (2009)

GeoNames, GeoNames web services for OSM (2010), Reterived from http://www.geonames.org (accessed 2010-09-17)

GeoVector (2010). GeoVector world surfer. Retrieved from http://www.geovector.com/applications/world-surfer/ (accessed 2010-09-17)

Goodchild, M.F.: Citizens as sensors: the world of volunteered geography. GeoJournal 69, 211–221 (2007)

Google AJAX, Google AJAX search API (2010), Retrieved from: http://code.google.com/apis/ajaxsearch/ (accessed 2010-09-17)

Haklay, M., Weber, P.: OpenStreetMap: User-generated street maps. IEEE Pervasive Computing, 12–18 (2008)

Haklay, M.M.E., Ellul, C.: Completeness in volunteered geographical information – the evolution of OpenStreetMap coverage in England (2008-2009). Journal of Spatial Information Science (2010)

Jacob, R., Zheng, J., Ciepluch, B., Mooney, P., Winstanley, A.C.: Campus guidance system for international conferences based on openStreetMap. In: Carswell, J.D., Fotheringham, A.S., McArdle, G. (eds.) W2GIS 2009. LNCS, vol. 5886, pp. 187–198. Springer, Heidelberg (2009)

Mountain, D., MacFarlane, A.: Geographic information retrieval in a mobile environment: evaluating the needs of mobile individuals. Journal of Information Science 5, 515–530 (2007)

OSM, OpenStreetMap (2010), http://www.openstreemap.org (accessed 2010-09-17)

Ravada, S., Kazar, B.M., Kothuri, R.: Query processing in 3D spatial databases: Experience with Oracle Spatial 11g. 3D Geo-Information Sciences, 153–173 (2009), doi:10.1007/978-3-540-87395-2

Rubin, A.: Geo/Spatial Search with MySQL (2006), Retrieved from: http://www.scribd.com/doc/2569355/Geo-Distance-Search-with-MySQL (accessed 2010-09-17)

Schöning, J., Krüger, A., Cheverst, K., Rohs, M., Löchtefeld, M., Taher, F.: PhotoMap: using spontaneously taken images of public maps for pedestrian navigation tasks on mobile devices. ACM, Bonn (2009)

Shek, S.: Next-generation Location-Based Services for mobile devices. CSC Grants (2010), Retrieved from http://asset-s1.csc.com/lef/downloads/ (accessed 2010-09-17)

Simon, R., Fröhlich, P.: A mobile application framework for the geospatial web. In: Proceedings of the 16th International Conference on World Wide Web, pp. 381–390. ACM, Banff (2007)

Simon, R., Fröhlich, P., Obernberger, G., Wittowetz, E.: The Point to Discover GeoWand. In: Krumm, J., Abowd, G.D., Seneviratne, A., Strang, T. (eds.) UbiComp 2007. LNCS, vol. 4717. Springer, Heidelberg (2007)

Skyhook, Skyhook location positioning, context and intelligence (2010), Retrieved from: `http://www.skyhookwireless-.com/` (accessed 2010-09-17)

Takeuchi, K., Kennelly, P.: iSeismometer: A geoscientific iPhone application. Computers & Geosciences 36, 573–575 (2010)

tgws, TurboGears Web Services (2010), Retrieved from: `http://code.google.com/p/tgws/ (accessed 2010-09-17)`

Willis, K., Holscher, C., Wilbertz, G., Li, C.: A comparison of spatial knowledge acquisition with maps and mobile maps. Computers, Environment and Urban Systems 33, 100–110 (2009)

Zhou, Z., Zhou, B., Huang, Q.: Evaluating query performance on object-relational spatial databases. In: 2nd IEEE International Conference on Comptuer Science and Information Technology, ICCSIT 2009, pp. 489–492 (2009), doi:978-1-4244-4250

Delimiting Imprecise Regions with Georeferenced Photos and Land Coverage Data*

Bruno Martins

Instituto Superior Técnico INESC-ID
Av. Professor Cavaco Silva, 2744-016 Porto Salvo, Portugal

Abstract. This paper presents an automated method for defining the boundaries of imprecise regions with basis on publicly available data. The method combines interpolation from a set of points which are assumed to lie in the region to be delineated, obtained from Flickr and evaluated through Kernel Density Estimation, with heuristics for refining the results that leverage on land coverage datsets obtained through remote sensing, integrated through an approached based on region shrinkage. The overall approach is evaluated by means of statistical classification measures, using regions whose boundaries are well defined. Our results shows that the method proposed here performs better than previous approaches described in the litterature, based solely on interpolation through Kernel Density Estimation.

1 Introduction

A major problem in the context of Geographic Information Systems relates to the vague and imprecise nature of the names and concepts that are often employed by the users of these systems. Geographic names such as *The Alps* or *Downtown Lisbon* have no limit geospatial boundaries that are officially accepted. Thus, these names can be interpreted differently by different people, although there is some consensus regarding their conceptualization. Since geographically vague names are used very often, the development of intelligent techniques for their interpretation is a research problem that has been gaining increasing interest. Automated approaches for delimiting geographic regions can nowadays have many practical applications, including the descovery of the boundaries of the areas of interest of Web resources, with basis on the geospatial locations of the visitors, or the augmentation of existing gazetters with geospatial footprints for vaguely defined places.

This papers describes a novel approach for automatically discovering the geospatial boundaries associated with vague regions, using datasets publicly available on the Web such as land coverage images obtained through remote sensing, or collections of points associated with region names, obtained by querying the API of the Flickr photo sharing website[1] (i.e., collections of geospatial

* This work was partially supported by the Fundação para a Ciência e a Tecnologia (FCT), through project grant PTDC/EIA-EIA/109840/2009 (SInteliGIS).

[1] http://flickr.com/services/api/

K. Tanaka, P. Fröhlich, and K.-S. Kim (Eds.): W2GIS 2011, LNCS 6574, pp. 219–229, 2011.

coordinates corresponding to points where specific the photos were taken). The proposed approach works by (i) aquiring points belonging to the vague region by querying Flickr, (ii) pre-processing the points in order to minimize the impact of photos that are incorrectly georeferenced, (iii) estimating the region for the vague area with basis on Kernel Density Estimation (KDE), and (iv) refining the region estimates through land coverage datasets, using an approach based on region shrinkage in order to approximate the boundaries, originally over-estimated through KDE, to regions of discontinuity in terms of land coverage. The proposed method was compared against a baseline that simply uses Kernel Density Estimation, taking as basis a set of names corresponding to specific regions for which we know the true geospatial limits. The obtained results show that the refined method proposed here indeed performs better than using KDE alone, resulting in an improvement of approximately 10% in terms of the F_1 metric.

The rest of this paper is organized as follows: Section 2 describes the most important related work. Section 3 presents the proposed method, detailing the aproaches used in pre-processing, the original KDE-based region estimation and region refinement through land coverage data. Section 4 describes the experimental validation of the proposed method, presenting the test data, the evaluation metrics, and the obtained results. Finally, Section 5 summarizes our conclusions and points directions for future work.

2 Related Work

The delineation of vague geographic concepts has been widely studied in Geographical Information Sciences. Previous work has explored methods for delimiting imprecise regions with basis on knowledge about points which are known to be inside or outside of the regions to be delineated, often assuming that names for vague regions co-occurr frequently with other placenames. Different sources of information have been experimented with, including user questionnaires [17], information about points and geospatial relations described over maps and gazetteers [1], textual information published on the Web [2,14], or georeferenced photographs published on sites such as Flickr [11].

One way of aquiring knowledge about the geospatial limits associated with vague regions is through the use of direct inquires with human experts [17], asking them to delineate the regions over a map, or asking them to classify specific points as being inside or outside the vague regions that we wish to delimit. Clough and Pasley, for instance, used questionnaires to define regions such as *Sheffield City Centre* or *The Peak District* (a rural area near *Sheffield*), showing photos to individuals and asking them to identify the places in the photos as belonging or not to the vague region [6]. Afterwards, these authors explored a method based on Kernel Density Estimation to find an approximation of the area corresponding to the vague region with basis on geospatial coordinates associated to the photos indicated by the subjects as belonging to the region. Altough metodologies based on questionnaires can obtain results of acceptable quality, they are particularly susceptible to scalability issues. Given the large amounts of georeferenced data that nowadays exists on the Web, it seems more promissing to use these data as an alternative to questionnaires.

Authors like Jones et al. described methods based on the use of Web search results for queries involving the names of the vague regions [14]. These approaches are based on the recognition and disambiguation of place names that co-occur with the names associated with the vague regions. Through the use gazetteers and text mining heuristics [15], place names are disambiguated into their corresponding geospatial coordinates. Each of these place names is also weighted according to their occurence frequency in the search results. Geometric techniques based on Voronoi triangulations or α-shapes [2] (i.e., a computer geometry algorithm that estimates the region defined by a set of points [19]), or techniques based on Kernel Density Estimation [14], are then used to discover the boundaries of the regions. In particular, Arampatzis et al. described a refined approach which uses patterns over the search engine queries in order to collect points that can be classified as inside or outside the vague region [2]. Geometric algorithms based on α-shapes and Voronoi triangulations are then applied to delineate the region, using both the interior and exterior points.

The approaches described in the previous paragraph require the recognition and disambiguation of place names mentioned in texts, a task which is error prone and computationally demanding. Other authors have explored similar approaches, although supported in data sources on which the acquisition process is less demanding, including georeferenced photographs on photo sharing sites like Flickr [11]. In fact, Flickr data has been used in a wide range of geospatial applications [7], given the abundance of direct links between geospatial coordinates (i.e., the locations where photos were taken, automatically acquired through GPS devices in cameras or mobile phones, or manually associated by the authors), calendar dates (i.e., the moments when the photos were acquired) and textual descriptions (i.e., titles, descriptions and tags associated to the photos).

Grothe and Schaab described two methods for the automatic delineation of vague regions with basis on Flickr, including a method based on Kernel Density Estimation and another based on One-Class Support Vector Machines [11], i.e. a variant of the classical Support Vector Machines classification approach requiring only positive examples [18]. The same authors have also described techniques for the optimization of the parameters required by the algorithms used for delimiting the regions. In our opinion, this publication is the one that best represents the current state of the art in the automatic delineation of imprecise regions, even though in terms of industrial applications, it is also worth mentioning the work described by a team of engineers from Flickr[2] which used an approach based on α-shapes to delineate the regions corresponding more than 150,000 names described in a large world-wide gazetteer.

The main contribution of this paper lies in the introduction of a refinement technique that, taking as input a density surface produced through an approach such as the ones presented by Grothe and Schaab, together with a

[2] `http://code.flickr.com/blog/2008/10/30/the-shape-of-alpha/`

detailed land coverage image where pixels are assigned a value that corresponds to the caracteristics of the terrain (e.g., water, vegetation, desert, urban area), is able to improve the region estimates according to discontinuities in the terrain (e.g., the borders of the vague region are likely to correspond to land masses, and pixels of water should not be included). The evaluation in most previous work was also made exclusively with vague areas corresponding to large regions, whereas we used a more diverse set of reference regions to evaluate the proposed method. Finally, this paper also presents experiments involving the use of pre-processing techniques for removing outliers from the set of points aquired from Flickr.

3 Delimiting Vague Regions

The method proposed in this paper for the delineation of vague regions uses the Flickr API as a way to collect data points that are known to be contained within the vague region. The method involves four steps, namely i) collecting the points, ii) pre-processing the points, iii) estimate the boundaries of the region, and iv) refine the initial boundary estimate.

In the first step, a query is made to the Flickr API using the standard UNIX *wget* tool, in order to obtain all coordinates associated to photos whose title, description of tags contain the name of the vague region. A maximum of 2.500 points is collected for each region.

In the pre-processing step, and in order to increase the variability in the data and decrease the impact of errors related with incorrect coordinates associated with the photos, the exact duplicate points are discarded. In a further attempt to eliminate errors, we also experimented with outlier removal methods. Specifically, the adaptive technique proposed by Filzmoser et al. was tested [9]. This technique considers the distance to the centroid point data, as well as the shape of the region containing the points, looking for outliers with basis on points that have a high Mahalanobis distance [16] towards the centroid point.

In the third and fourth steps, the regions are delimited with basis on the corresponding points, using a Kernel Density Estimation method, and afterwards refined with basis on a land coverage image, though an approach based on region shrinkage. The rest of this section details both these steps.

The methods for outlier removal, Kernel Density Estimation, and region refinement were all implemented in the *R system*[3] statistical environment. *R system* allows the rapid realization of experiments, exporting the results as image representations for the regions (e.g., in the *PNG* or *PDF* formats) or as *GML*, *KML* ou *ESRI Shapefile* documents for posterior analysis with other Geographical Information Systems. The method for Kernel Density Estimation is available from the np[4] extension package [13]. The method for outlier removal is available from the mvoutlier[5] extension package [16].

[3] `http://www.r-project.org/`
[4] `http://cran.r-project.org/web/packages/np`
[5] `http://cran.r-project.org/web/packages/mvoutlier`

3.1 Delimiting Vague Regions With Kernel Density Estimation

A technique from spatial statistics for the estimation of density surfaces, commonly refered to as Kernel Density Estimation, has been widelly used in Geographical Information Sciences as a way to estimate the density distribution of geospatial processes that are at the origin of certain sets of discrete observations [4]. In our specific application, and assuming that the points that are associated with the name of the vague region correspond to point observations that are contained within the region, this technique can be used as a way to estimate the probability of each point within a bi-dimentional surface belonging or not to the interior of the vague region [11].

For a set of points $X = \{x_1, ..., x_n\}$, the function bellow returns the estimated density for a point p, with basis on a Gaussian kernel. In the formula h is a suavization parameter that controls the resolution of the kernel. Computing the density surface with the formula shown bellow is usually done through a grid that discretizes the bi-dimentional space associated with the surface.

$$f(p) = \frac{1}{|X| \times h} \sum_{x_i \in X} \frac{1}{\sqrt{2\pi}} e^{\frac{(p-x_i)^2}{2h^2}} \tag{1}$$

The use of Kernel Density Estimation for the automatic discovery of the limits associated with a vague region requires a second parameter, corresponding to a cut value b. The contours of the vague region correspond to a contour line over the density surface, chosen in a way to include all points with a density b.

The choice of parameters h and b is very important, since using the same set of points with different h and b values results in very different estimations for the vague region. There are known methods for automatically choosing the h parameter with basis on the input data [4,13]. The R statistical environment, through the *np* extension package, offers an implementation of the Kernel Density Estimation technique that supports various kernel functions and methods for automatically setting the h parameter. This software was used in the experiments reported in this paper. The value for the parameter b was chosen empirically with basis on experiments.

3.2 Refining the Density Estimates through Region Shrinkage

Initial experiments with the Kernel Density Estimation method, which was described in the previous section, revealed that the method tended to over-estimate the regions associated to the vage areas. Under the assmumption that the real limits of a vague area often correspond to areas of discontinuity in the terrain (e.g., the vague areas are often delimited by the trajectory of rivers, by mountain ranges, deserts, of regions of dense vegetation, etc.), we propose a method for refining the initial areas estimated through the KDE method by shrinking them according to the nearest areas of discontinuity in terms of the characteristics of the terrain. These caracteristics can be obtained from one of the nowadays very popular land coverage image datasets, where each pixel is assigned to a code that corresponds to the caracteristics of the terrain (e.g., a diffent code for water, vegetation, desert, urban areas, etc.). Specifically, in the experiments reported in this paper, we used a dataset from the Global Land Cover Facility[6] that consists of a map image of fourteen different land cover categories for the entire Earth, in a resolution of 1 kilometer per pixel, build from imagery from the NOAA-AVHRR satellites acquired between 1981 and 1994 [12].

In the area of Computer Vision, region growing is a simple and very popular region-based image segmentation method, which works by iteratively examining neighboring pixels of initial seed points and determining whether the pixel neighbors should be added to the region [20]. Specifically, this method takes as input an image together with a set of seeds, corresponding to the points inside a particular region of the image that we wish to discover. The region is then iteratively discovered by comparing all unallocated neighbouring pixels to the others, in order to find contiguous areas. This process continues until all the corresponding pixels are allocated to the region.

Our algorithm for adjusting the KDE estimates follows similar ideas, iteratively shrinking an area instead of growing it. We examine the neighboring pixels of the points corresponding to the boundary of the region, determining if these points should indeed bellong to the region or not. The pseudocode for the algorithm is shown bellow:

1. Search the KDE surface for discontinuity pixels and add them to a set S.
2. While the set S is not empty proceed by:
   (a) Taking a pixel P from the set S.
   (b) Examining the land-coverage of the 8-connected neighbors of P.
   (c) If an inner-neighbour P_2 of P has the same land-coverage of P then:
      i. Adjust the density value of P to a value bellow the threshold b .
      ii. Add P_2 to the set S.

In the first step, the discovery of the pixels corresponding to areas of discontinuity is based on searching the KDE surface to find the pixels whose oposing neighbors (e.g., left and right pixels, top and bottom pixels, north-west and south-east pixels, etc.) have distinct values in terms of the estimated density (i.e., one of the pixels has a higher density then the threshold b and the other has a density value lower than the threshold). A similar procedure is used in step 2.(b), by checking if the neighbour whose density is lower than the threshold corresponds to a land-coverage class that is shared with the other pixel.

Iteratively, the above procedure shrinks the over-estimated region that is produced by the Kernel Density Estimation method, readjusting the boundaries in a way that the limits for the vague region are made to correspond to areas of discontinuity in terms of the land coverage for the terrain.

4 Experimental Validation

To validate the proposed method for delimiting vague regions, experiments were made envolving the names corresponding to 18 regions whose true geospatial

[6] http://www.landcover.org/data/landcover/

Table 1. Results obtained for the different vague regions

	num. points	KDE with all points			KDE without outliers			refined approach		
		P	R	F_1	P	R	F_1	P	R	F_1
Albania	1336	0.080	1.000	0.149	0.695	0.841	0.761	0.814	0.809	**0.811**
Belarus	1130	0.130	0.946	0.229	0.867	0.592	0.704	**0.870**	0.430	0.575
Croatia	1829	0.130	0.999	0.230	0.368	0.432	0.397	0.575	0.414	**0.481**
France	1448	0.260	0.971	0.410	0.536	0.591	0.562	0.748	0.844	**0.793**
Germany	2297	0.224	0.999	0.366	6.107	0.136	0.267	0.633	0.523	**0.573**
Greece	2015	0.094	0.998	0.173	0.294	0.876	0.440	0.580	0.827	**0.682**
Ireland	2360	0.147	0.999	0.257	0.520	0.886	0.656	0.771	0.706	**0.737**
Italy	1593	0.190	0.995	0.319	0.481	0.905	0.628	0.670	0.909	**0.771**
Luxembourg	1633	0.017	1.000	0.034	0.278	0.991	0.435	0.309	0.987	**0.471**
Switzerland	1867	0.153	0.999	0.265	0.700	0.990	0.820	0.723	0.988	**0.835**
Ukraine	1110	0.214	0.950	0.349	0.687	0.290	0.585	0.700	0.646	**0.672**
Cornwall	1708	0.110	0.970	0.197	0.703	0.984	0.821	0.934	0.942	**0.938**
Durham	1232	0.000	0.999	0.001	0.408	0.999	0.580	0.625	0.916	**0.743**
Essex	1449	0.102	0.989	0.186	0.730	0.999	0.843	0.821	0.984	**0.895**
Warwickshire	972	0.485	0.999	0.653	0.907	0.978	0.941	0.952	0.939	**0.946**
West Midlands	1211	0.202	0.999	0.337	0.887	0.996	0.938	0.949	0.983	**0.966**
West Sussex	838	0.585	0.985	0.735	0.926	0.916	0.921	0.977	0.877	**0.924**
West Yorkshire	1958	0.185	0.999	0.313	0.722	0.999	0.838	0.790	0.982	**0.876**
Average	1555	0.184	0.989	0.289	0.934	0.800	0.674	0.747	0.817	**0.761**
St. Deviation	441	0.146	0.018	0.183	1.307	0.272	0.202	0.169	0.193	**0.158**

limits were known in advance. The choice of regions was made in an attempt to ensure a large diversity in terms of area, shape, and type of the region, thus supporting tests with diverse conditions. The 18 considered regions include the names of 11 European countries that were used in the previous work by Grothe and Schaab [11], together with the names of 7 other smaller regions. The quality of the produced regions was evaluated by means of statistical classification measures such as Precision or Recall. For each vague geographical name, let D be its true region and let D^* be the produced estimate. Precision P and Recall R can be defined according to the formulas bellow:

$$P = \frac{Area(D \cap D^*)}{Area(D)} \tag{2}$$

$$R = \frac{Area(D \cap D^*)}{Area(D^*)} \tag{3}$$

In this paper, the majority of results are reported in terms of the F_1 metric, which summaries Precision and Recall through their harmonic mean.

$$F_1 = \frac{2 \times P \times R}{P + R} \tag{4}$$

Table 1 presents the obtained results for each of the considered vague regions, separatly presenting measurements for (i) a technique based on Kernel Density Estimation considering all points, (ii) a technique based on Kernel Density Estimation in which outlier points have been removed, and (iii) the refined technique presented in this paper. In all experiments, the Kernel Density Estimation was computed with a b threshold value of 1×10^{-6}. Table 1 also presents the number of points gathered for each of the considered regions.

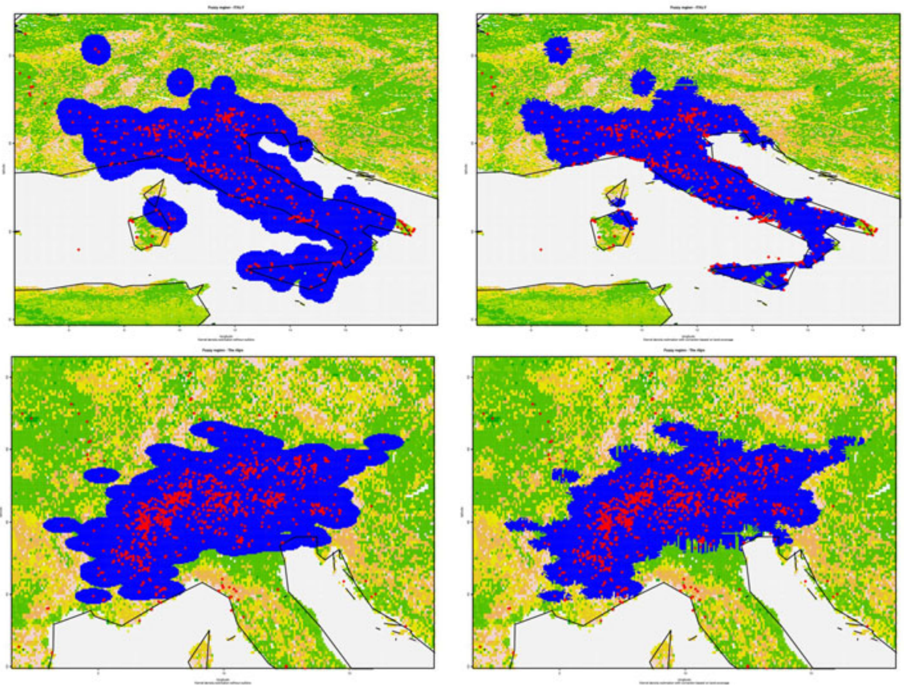


Fig. 1. Results obtained for *Italy* and *The Alps*

The results in Table 1 show that the proposed refined approach indeed produces a substantial improvement over a Kernel Density Estimation baseline. On average, this improvement corresponds to a difference of approximately 10% in terms of the F_1 measure. Moreover, the results also show that the pre-processing procedure for removing outlier points also improves the results.

In the case of names for the 11 European countries that had been previously used in [11], our results show average F_1 scores of 0.25, 0.57 and 0.67, respectively for the methods corresponding to the usage of KDE, KDE removing outliers, and KDE refined through land coverage data. These results are somewhat inferior to those reported in [11], which corresponded to average F_1 scores of 0.69 in the case of a method based on KDE, and 0.74 for a method based on one-class SVMs. Still, the experimental conditions (i.e., the points collected for the different regions) were somewhat different in our case, and we also did not try to optimize the bandwidth parameter associated with the KDE method through neither heuristic approaches or cross-validation procedures. It is our belief that the approach proposed here, which used land coverage data to refine the estimates produced by a density estimation method, could also be used to improve the results of the one-class SVM method used by Grothe and Schaab [11].

In Figure 1, we illustrate the results obtained for two distinct regions, namelly *Italy* and *The Alps*.

5 Conclusions and Future Work

This paper presented an automated method for delimiting imprecise regions with basis on publicly available data. The method combines estimation from a set of points which are assumed to lie in the region, obtained by querying Flickr and interpolated from the set of points through a Kernel Density Estimation method, with heuristics for refining the results that leverage on land coverage datsets obtained through remote sensing, integrated through an approached based on region shrinkage. The overall approach for finding region boundaries was evaluated by means of statistical classification measures, using a set of 18 regions whose boundaries are well defined. Our results show that the refined method performs better than simpler methods described in the litterature, based solely on interpolation through Kernel Density Estimation.

Despite the good results, there are also many interesting challenges for future work. Intuitively, the proposed algorithm will work well only when the cognition for a given region by the majority of people is not wrong. Indeed, the strongest interest point of the proposed approach is to be able to associate a given place name with the geospatial region that the majority of people would also associate to it. However, it is not clear how the proposed method will cope with the case of regions that are often missunderstood by people. Future experiments with the proposed method, using a larger set of regions and possibly also using human experts, should try to address this issue.

There is also a huge literature on the usage of Kernel Density Estimation in related applications. It would be interesting to experiment with methods for setting the bandwidth h parameter dynamically, as well as with different kernels [10] and different density estimation techniques. Of particular interest is the previous work by Carlos et al., where the authors described a density estimation method that adapts the bandwidth parameter with basis on the underlying population distribution, arguing that when the effect of the background population is sucessfully accounted for, the differences in point patterns over similar population areas become more dicernible [5]. For future work, we plan on experimenting with the usage adaptive bandwith approaches based on datasets such as the LandScan Global Population Database[7] or the Gridded Population of the World[8]. Moreover, besides population density and land coverage datasets, there are many other sources of geographical information that could be used to refine the estimates for the vague areas (e.g., road networks data, satellite imagery, etc.).

Another interesting challenge for future work is related to the evaluation of the obtained regions. The metrics of Precision and Recall used in our experiments effectively allow one to draw conclusions regarding the quality of the produced regions. However, particular applications may be more or less sentitive to the quality of the region approximations, instead being interested in the preservation or properties such as tolological relations between the generated regions [8]. For

[7] `http://www.ornl.gov/sci/landscan/`

[8] `http://sedac.ciesin.columbia.edu/gpw/`

future work, it would be interesting to measure the effect the quality of the region approximations has on particular algorithms that use them, in applications such as Geographic Information Retrieval [3].

References

1. Alani, H., Jones, C.B., Tudhope, D.: Voronoi-based region approximation for geographical information retrieval with gazetteers. International Journal of Geographical Information Science 15(4) (2001)
2. Arampatzis, A., van Kreveld, M.J., Reinbacher, I., Jones, C.B., Vaid, S., Clough, P., Joho, H., Sanderson, M.: Web-based delineation of imprecise regions. Computers, Environment and Urban Systems 30(4) (2006)
3. Jones, C.B., Purves, R.S.: Geographical information retrieval. International Journal of Geographical Information Science 22(3) (2008) (editorial article)
4. Brunsdon, C.: Estimating probability surfaces for geographical point data: an adaptive kernel algorithm. Computers and Geosciences 21(7) (1995)
5. Carlos, H.A., Shi, X., Sargent, J., Tanski, S., Berke, E.M.: Density estimation and adaptive bandwidths: A primer for public health practitioners. International Journal of Health Geographics 9(39) (2010)
6. Clough, P., Pasley, R.: Images and perceptions of neighbourhood extents. In: Proceedings of the 6th ACM Workshop on Geographial Information Retrieval (2010)
7. Crandall, D.J., Backstrom, L., Huttenlocher, D., Kleinberg, J.: Mapping the world's photos. In: Proceedings of the 18th International Conference on World Wide Web (2009)
8. Deng, M., Chen, X., Kusanagi, M., Phien, H.N.: Reasoning of topological relations between imprecise regions. Annals of GIS 10(1) (2004)
9. Filzmoser, P., Garrett, R.G., Reimann, C.: Multivariate outlier detection in exploration geochemistry. Computers and Geosciences 31 (2005)
10. Gibin, M., Longley, P., Atkinson, P.: Kernel density estimation and percent volume contours in general practice catchment area analysis in urban areas. In: Proceedings of the GIScience research UK Conference (2007)
11. Grothe, C., Schaab, J.: Automated footprint generation from geotags with kernel density estimation and support vector machines. Spatial Cognition and Computation 9(3) (2009)
12. Hansen, M., DeFries, R., Townshend, J.R.G., Sohlberg, R.: Global land cover classification at 1km resolution using a decision tree classifier. International Journal of Remote Sensing 21 (2000)
13. Hayfield, T., Racine, J.S.: Nonparametric econometrics: The np package. Journal of Statistical Software 5(27) (2008)
14. Jones, C.B., Purves, R.S., Cloughs, P.D., Joho, H.: Modelling vague places with knowledge from the web. International Journal of Geographical Information Science 22(10) (2008)
15. Leidner, J.: Toponym Resolution in Text: Annotation, Evaluation and Applications of Spatial Grounding of Place NamesPhD thesis, Institute for Communicating and Collaborative Systems, School of Informatics, University of Edinburgh (2007)
16. Mahalanobis, P.C.: On the generalised distance in statistics. Proceedings of the National Institute of Sciences of India 2 (1936)

17. Montello, D., Goodchild, M.F., Gottsegen, J., Fohl, P.: Where's downtown?: behavioural methods for determining referents of vague spatial queries. Spatial Cognition and Computation 3(2&3) (2003)
18. Munoz, A., Moguerza, J.M.: Estimation of high-density regions using one-class neighbor machines. IEEE Transactions on Pattern Analysis and Machine Intelligence 28(3) (2006)
19. Pateiro-López, B., Rodríguez-Casal, A.: Generalizing the convex hull of a sample: The r package alphahull. Journal of Statistical software 34(5) (2010)
20. Rolf, A., Leanne, B.: Seeded region growing. IEEE Transactions on Pattern Analysis and Machine Intelligence 16(6) (1994)

Visualization of Relationships among Historical Persons Using Locational Information

Takahiko Osaki[1], Sho Itsubo[1], Fuminori Kimura[2], Taro Tezuka[2], and Akira Maeda[2]

[1] Graduate School of Science and Engineering
[2] College of Information Science and Engineering
Ritsumeikan University
{cm003060,cm001061}@ed.ritsumei.ac.jp,
fkimura@is.ritsumei.ac.jp,
{tezuka,amaeda}@media.ritsumei.ac.jp

Abstract. The use of text mining for obtaining new knowledge from historical documents has gained wide attention in humanities research. We propose a method of revealing and visualizing relationships between historical persons using locational information expressed in historical documents. For each person, a vector of co-occurring place names is constructed. We then clustered persons using this vector in the K-means algorithm. We conducted experiments using *Hyohanki*, a diary written by an aristocrat in the classical era. The result showed that the obtained clusters match well with historically meaningful groups, indicating the effectiveness of our proposed method.

Keywords: Historical documents, Text mining, Visualization.

1 Introduction

In recent years, there is an increasing use of digital technology in the study of humanities. Many historical documents are now digitally archived, enabling further analysis using computers. There are archives that are accessible on the World Wide Web, including Union Catalog of the Collections of the National Art Museums, Japan [1] and Perseus Digital Library [2].

Until recently, the storage of historical documents and data has been the main target of digital archive research. There are, however, many works that go on to analyze the content of the historical documents using text mining techniques. In this paper, we propose a method of visualizing relationships among historical persons using locational information appearing in documents. Specifically, we use the digitized text of *Hyohanki* as a data source. *Hyohanki* is a personal diary written by Nobunori Taira (A.D.1112-1187) in late Heian period of Japan. It is said that he has written his diary for 52 years (from 1132 to 1184), but today we only have its fair copy (written by himself) dating from 1132 to 1171. Although some part of it has been deteriorated and missing, all of the existing pages are digitized into text format. The existing pages consist of 2,488 diary entries.

Nobunori Taira worked at the imperial court and served as a trusted vassal of the emperors and ministers. It means he was at a position to know events occurring around

K. Tanaka, P. Fröhlich, and K.-S. Kim (Eds.): W2GIS 2011, LNCS 6574, pp. 230–239, 2011.

the center of power of the time. Unlike a modern day personal diary, *Hyohanki* is partly a record of official events and political actions, making it a rich source for knowing policies, disputes and conspiracies in the court. It is therefore a document of high importance.

Figure 1 is an example of the original copy of *Hyohanki* [3]. In analysis, we used digitized text created by Sugihashi Laboratory in the College of Literature, Ritsumeikan University, based on the published version [4].

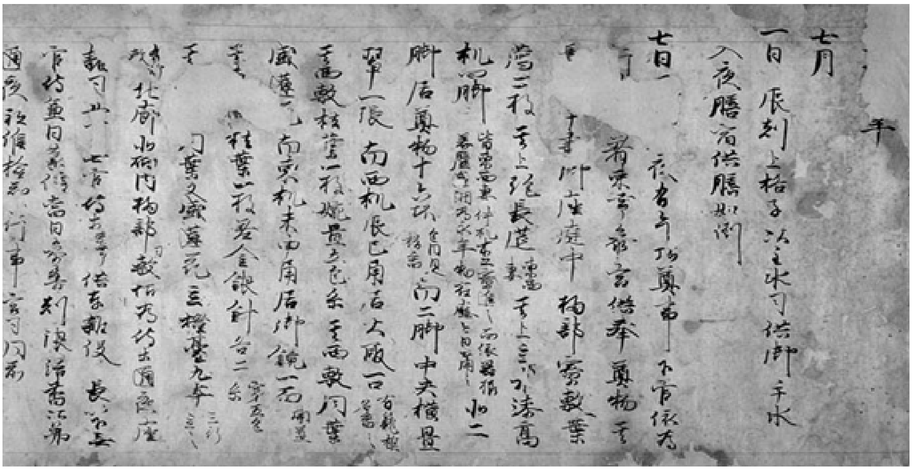


Fig. 1. Example of the original copy of a historical Japanese diary, *Hyohanki*

We used place names appearing in the document to characterize each person's activity. It is expected that a person's characteristics can be expressed based on activities, rather than statistic features such as social classes or affiliation. We defined similarity between persons using co-occurring place names. Relationships among persons were visualized based on this similarity measure.

The rest of the paper consists of the following sections. Section 2 gives related work. Section 3 describes our method in detail. Section 4 describes the result of evaluation. Section 5 is the conclusion.

2 Related Work

2.1 Mining Historical Documents

Until now, there has been only a small number of works that proposes extended search and analysis functions for digital archives of historical documents. Gerlach and Fuhr proposed "cross-age search systems" that enables searching of ancient document using a modern language [5]. Khaltarkhuu and Maeda proposed such system for traditional Mongolian texts [6]. Kimura and Maeda proposed an extended search system for historical documents which enables to search Hyohanki written in ancient Japanese using a modern Japanese query [7].

Recently, Yoshida visualized walking paths of the 12th century aristocrats when they walked through Kyoto, the capital of Japan back then [8]. It significantly illustrated their daily activities and differences among social classes and occupations. While Yoshida's work indicates the effectiveness of using locational information for visualizing activities of historical persons, it involved intensive work since extraction of place names from the documents were performed manually. One of the aims of this paper is to enable such elaborate analysis automatically by the use of text mining and data mining techniques.

2.2 Spatial Clustering Analysis and Visualization

Spatial clustering has been applied to different domains of geographical information science and proved to be effective[9,10]. There are also research on visualizing relationships obtained from documents using graphs[11,12]. In our paper, we utilize graphs to visualize information obtained from historical documents, extending this approach to a new field.

Kazama et al. extracted relationships among persons based on co-occurrences in web pages. They considered personal relationships as a new type of link structure similar to hyperlinks on the web, and applied various mining techniques on the graph[13]. We perform similar graph mining techniques, but we do not use direct co-occurences between persons' names. Instead, we use indirect relationships based on place names.

3 Method

3.1 Information Extraction

In the first step of our proposed method, we obtain frequencies of co-occurrences between each person's name and place names. Since *Hyohanki* is written in classical Japanese, we cannot attach part-of-speech tags using existing morphological analyzers that were trained using modern Japanese. We therefore used pattern matching to find place names that were included in the dictionary. We used the "Index of Kyoto's Place Names" created by Noboru Tani based on the Outline of Heiankyo [14] and Japan's Historical Place Names 27: Place Names of Kyoto [15] as data sources for place names.

We use a co-occurrence as the indicator of a relationship between a person and a location. If a person's name and a place name appear in a same paragraph, we consider it as a co-occurrence between the two. In many cases, a co-occurrence between a person A and a location B indicates that a person A was at a location B. There are, of course, cases where A and B co-occur in an expression such as 'A has never visited B." It is difficult to exclude such cases, unless applying extensive natural language processing. If we could collect many co-occurrences, however, the result is expected to represent a positive relationship between a person and a place name, due to the law of large numbers.

Since each paragraph in Hyohanki usually describes a specific scene, we considered it to be a better unit than dates. Further refinement might be possible if we adjusted temporal granularity, but that is a part of future work. In this paper we use a paragraph as an approximation.

3.2 Clustering

Each location name is considered as a dimension of a vector space. For each person, we create a vector having the number of co-occurrences with a place name as a component. For the similarity measure, we used cosine similarity.

In clustering, there is a proposal of improving convergence by setting initial values in a probabilistic manner [16]. We used this modified K-means method for clustering persons' names. Parameters in our method are K and L. K is the number of clusters used in K-means clustering. L is the number of repetitions for finding the optimal initial centroids. The algorithm is as follows.

1. Randomly select initial value x, indicating a person.
2. Select x' that maximizes $p(x') = \frac{D(x')^2}{\sum_y D(y)^2}$ and add it as a new centroid of a cluster. ($D(x')$ and $D(y)$ indicate the distances from x' and y to the nearest cluster, respectively. We use the value subtracting 1 from the similarity as the distance).
3. Repeat step 1 and 2 until K initial clusters are obtained.
4. Allocate each vector to its nearest cluster. (The distance between a vector and a cluster is defined by the distance between a vector and the centroid of the cluster).
5. Revise the coordinate of the centroid for each cluster.
6. Repeat step 4 and 5 until vectors are no longer reallocated.
7. Repeat step 1 to 6 for L times, and obtain the result that has the minimum total of the distance between each data and the centroid of the cluster it belongs to.

3.3 Visualization

We use the similarity measure and the result of clustering for visualization. We used JUNG, a Java open source library for drawing graph structure. It has a capability of drawing graphs when the relevance measures between each pair of nodes are given.

4 Experiments

Using our proposed method, we created graphs that visualize relationships between historical persons. We focused on the time range of Hougen Rebellion, starting in early July 1156 and ending in late July of the same year.

Hougen Rebellion is a short civil war caused by a power struggle between the Emperor Goshirakawa and the former Emperor Sutoku, who was the elder brother of Goshirakawa. As a result of the rebellion, two *samurai* warrior clans, Taira and Minamoto, gain power and the peaceful Heian period ruled by the aristocrats Fujiwara family ends. It marks the turning point from the classical era to the medieval era in the Japanese history.

We chose 78 persons belonging to either the faction following the former Emperor Sutoku or the faction following the Emperor Goshirakawa [17]. Most of them are aristocrats and samurai warriors. It is distinguishable from historical records to which faction each person belonged to. In *Hyohanki*, 31 of these persons had co-occurrence with location names. We used $K = 3$ for K-means clustering and $L = 20$ for initialization.

4.1 Visualizing Relationships among Persons

Figure 2 shows the result of visualization using the similarity of co-occurring location names. The shape of a node indicates to which faction each person belonged to. A circular node indicates that he followed the former Emperor Sutoku, and a square node indicates that he followed the Emperor Goshirakawa. Lines are drawn when similarity is over 0.4. Dotted lines indicate similarity between 0.4 and 0.7. Solid lines indicate similarity over 0.7.

Figure 3 shows the result of clustering. The shape of the node indicates to which cluster the person was allocated to. The members of the cluster 1 is represented by triangular nodes, the members of the cluster 2 is represented by circular nodes, and the members of the cluster 3 is represented by square nodes. Table 1 shows to which cluster and to which faction each person belonged to.

The coordinates of each node in Figures 2-3 do not correspond to the actual geographical coordinates. The nodes are in a space of logical relationships, rather than in the actual space.

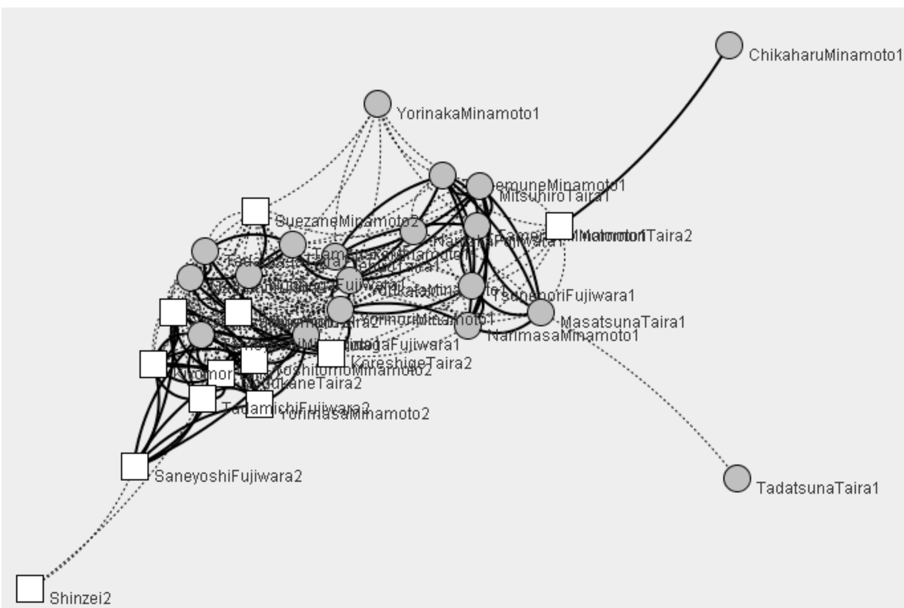


Fig. 2. Relationships among persons and historical factions

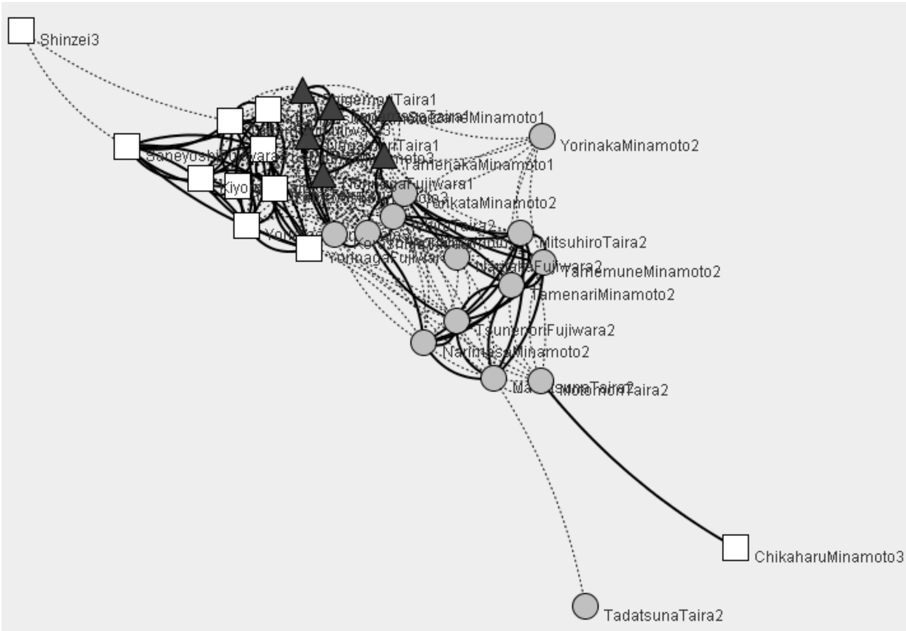


Fig. 3. Result of clustering persons using locational information

4.2 Analysis of Significant Place Names

The next step in our analysis was to characterize each cluster to see the meaning behind grouping of the persons. Since we used place names as features, clusters are characterized using place names. We defined the score $S(p, c)$ as follows.

1. Calculate cosine similarity $C(p, c)$ between the centroid vector for a cluster c and a place name vector p (which is an element of the basis).
2. Obtain score $S(p, c)$ by dividing $C(p, c)$ by the average for all c's, as follows. K is the number of clusters.

$$S(p, c) = \frac{C(p, c)}{\frac{1}{K}\sum_{c'=1}^{K} C(p, c')}$$

The value of $S(p, c)$ will range from 0 to K. Place names with high values of $S(p, c)$ are the ones that are specific to the cluster c. Table 2 shows the list of place names with high values of $S(p, c)$ for each cluster. The result contains several intuitive features. In cluster 2 (circle), a high score is assigned to “Takamatsuden Palace”, where the headquarters of the Emperor Goshirakawa’s faction was located at. In cluster 3 (square), a high score is assigned to “Shirakawa”, where the former Emperor Sutoku’s faction placed their headquarters. In cluster 1 (triangle), high scores are assigned to places where Sutoku’s faction seized from Goshirakawa’s faction, for example Higashisanjoden Palace.

Table 1. Comparison of factions and clusters

	Cluster 1 (triangle)	Cluster 2 (circle)	Cluster 3 (square)
faction of the former Emperor Sutoku	Nagamori Taira Norinaga Fujiwara Tamenaka Minamoto Tadamasa Taira	Narimasa Minamoto Tadatsuna Taira Yorikata Minamoto Yorinaka Minamoto Tamenari Minamoto Yorinori Minamoto Tsunenori Fujiwara Tadatsuna Taira Tamemune Minamoto Iehiro Taira Mitsuhiro Taira Naritaka Fujiwara	Yorinaga Fujiwara Chikaharu Minamoto
faction of the Emperor Goshirakawa	Suezane Minamoto Shigemori Taira Koreshige Taira	Motomori Taira	Shinzei Yoshitomo Minamoto Yorimasa Minamoto Tameyoshi Minamoto Saneyoshi Fujiwara Kiyomori Taira Yoshiyasu Minamoto Tadamichi Fujiwara Nobukane Taira

Table 2. Place names that signify each cluster

Cluster 1 (triangle)		Cluster 2 (circle)		Cluster 3 (square)	
Shougon'in Temple	2.157	Emonjin	2.832	Tokuchoujuin Temple	3.000
Byoudouin Temple	2.157	Koryuji Temple	2.832	Gokurakuji Temple	3.000
Tobaden Palace	1.772	Shirakawa	2.832	Sakyo	3.000
Higashisanjoden Palace	1.772	Houjuji Temple	2.152	Omono	3.000
Chisokuin Temple	1.772	Nankyo	1.977	Daishougun	3.000
Doushin	1.632	Ooeyama Mountain	1.763	Hon'in	3.000
Ouminokuni Prefecture	1.583	Kamomatsuri Festival	1.763	Hosshouji Temple	3.000
Saemonfu	1.463	Minonokuni Prefecture	0.845	Kizu	3.000
Minonokuni Prefecture	1.402	Byoudouin Temple	0.650	Shokomyoin Temple	3.000
Sanyodo Road	1.366	Shougon'in Temple	0.650	Hachijo	3.000
Ooeyama Mountain	0.975	Doushin	0.492	Nishinotouin Temple	3.000
Kamomatsuri Festival	0.975	Tobaden Palace	0.458	Nantei	3.000
Nankyo	0.596	Higashisanjoden Palace	0.458	Takamatsuden Temple	3.000

4.3 Discussion

The result shows that the obtained clusters significantly correspond to the historically known factions. The cluster 2 (circle) corresponds to the former Emperor Sutoku's faction, and the cluster 3 (square) correspond to the Emperor Goshirakawa's faction. The cluster 1 (triangle) is the intermediate group. This is a satisfactory result, considering the fact that we used only co-occurrences with place names and no other external information.

There were some exceptions, though. For example, Yorinaga Fujiwara, one of the main figures in the former Emperor Sutoku's faction, was allocated to the cluster consisting mostly of the members of the Emperor Goshirakawa's faction. To clarify the reason for such strange allocation, a further exploration of the raw data is required. A close investigation on frequently occurring pairs among persons' names and place names may reveal the reasons for such allocation.

5 Conclusion

We proposed a method of revealing and visualizing relationships among historical persons by focusing on place names appearing in digitized historical documents. We used cosine similarity and a modified K-means algorithm to create graphs and cluster persons.

In the experiments, we used persons that we know to which faction he belonged to during the Hougen Rebellion. The result showed a strong correspondence between the factions and the clusters, indicating effectiveness of using location information for clustering people.

In the future work, we aim to raise precision of clustering. In our experiments we used only a small number of persons. In such a case, the number of necessary clusters would be small, and the precision is high enough. Deciding the number of the clusters (or choosing a model) is not a big problem either. It will not be so in case where many persons are involved. We plan to use newly proposed variations of the K-means clustering, such as the division and merging algorithm [18]. Since one of the characteristics of a diary is that it contains information regarding time, we would utilize it also. Such approach will enable spatial-temporal analysis on historical person's activity.

There is a possibility that the functionality of a building or a place may change as time passes. If that happens, co-occurrences with a certain building may not indicate a relationship or similarity between two persons. Our present method does not consider such change in the functionality of a building or a place, yet since we have temporal information attached to each co-occurrence, in the future work we plan to cope with such a problem.

We also plan to use the official rank of each person. This is because the rank of each person is described very often in Hyohanki. Using persons' names, place names, and official ranks altogether, more features of the classical society can be revealed.

In visualization, we are intending to improve our interface by allowing the user to select a specific person and focus on visualizing relationships around him. It will enable investigation for specific purposes. On the other hand, by focusing on a specific set of place names, we are expecting to enable various types of clustering.

The method can be applied to other scenes within Hyohanki or to other texts. It is expected to humanities researchers in discovering new groups of persons during a significant historical event or a certain era.

Acknowledgments

This work was supported in part by the Grant-in-Aid for the Global COE Program "Digital Humanities Center for Japanese Arts and Cultures(DH-JAC)" from the Ministry of Education, Culture, Sports, Science and Technology (MEXT) of Japan, MEXT Grant-in-Aid for Strategic Formation of Research Infrastructure for Private University "Sharing of Research Resources by Digitization and Utilization of Art and Cultural Materials" (Grant Number: S0991041), MEXT Grant-in-Aid for Young Scientists (B) "Research on Information Access across Languages, Periods, and Cultures" (Leader: Akira Maeda, Grant Number: 21700271), and MEXT Grant-in-Aid for Young Scientists (B) "Object Identification System using Web Image Collection and Machine Learning" (Leader: Taro Tezuka, Grant Number: 21700121).

References

1. Union Catalog of the Collections of the National Art Museums, Japan, http://search.artmuseums.go.jp/
2. Perseus Digital Library, http://www.perseus.tufts.edu/
3. Digital Library of Kyoto University, http://edb.kulib.kyoto-u.ac.jp/minds.html
4. Domestic history lab, Literature department, Kyoto University, Hyohanki, Shibunkaku Publishing (1988) (in Japanese)
5. Gerlach, A.E., Fuhr, N.: Retrieval in text collections with historic spelling using linguistic and spelling variants. In: Proceedings of the 7th ACM/IEEE Joint Conference on Digital Libraries (JCDL 2007), Vancouver, Canada, pp. 333–341 (2007)
6. Khaltarkhuu, G., Maeda, A.: Retrieval technique with the modern Mongolian query on traditional Mongolian text. In: Sugimoto, S., Hunter, J., Rauber, A., Morishima, A. (eds.) ICADL 2006. LNCS, vol. 4312, pp. 478–481. Springer, Heidelberg (2006)
7. Kimura, F., Maeda, A.: An approach to information access and knowledge discovery from historical documents. In: Proceedings of Digital Humanities, College Park, Maryland, USA, pp. 359–361 (2009)
8. Yoshida, M.: Heiankyo visualized by the walking paths of aristocrats, In: 29th Ritsumeikan Global COE Seminar (2008) (in Japanese), http://www.arc.ritsumei.ac.jp/lib/GCOE/seminar/asx2/20080729yoshida.asx
9. Osaragi, T.: Spatial clustering method for geographic data, Working Paper, Centre for Advanced Spatial Analysis, University College London, Paper 41 (2002)
10. Jacquez, G.M.: Spatial cluster analysis. In: Fotheringham, S., Wilson, J. (eds.) The Handbook of Geographic Information Science, pp. 395–416. Blackwell Publishing, Malden (2008)
11. Toyoda, M., Kitsuregawa, M.: A system for visualizing and analyzing the evolution of the web with a time series of graphs. In: Proceedings of the 16th ACM Conference on Hypertext and Hypermedia (Hypertext 2005), Salzburg, Austria, pp. 151–160 (2005)
12. Matsuo, Y., Mori, J., Hamasaki, M., Ishida, K., Nishimura, T., Takeda, H., Hasida, K., Ishizuka, M.: POLYPHONET: an advanced social network extraction system from the web. In: Proceedings of the 15th International Conference on World Wide Web (WWW 2006), Edinburgh, Scotland, pp. 397–406 (2006)
13. Kazama, K., Sato, S., Fukuda, K., Murakami, K., Kawakami, H., Katai, O.: Evaluation of Using Human Relationships on the Web as Information Navigation Paths. In: Proceedings of the 1st International Workshop on Agent Network Dynamics and Intelligence (ANDI 2005), Kitakyushu, Japan, pp. 15–20 (2005)

14. Society for Ancient Studies, The Outline of Heiankyo. Kadokawa Publishing (1994)
15. Hayashiya, T., Murai, Y., Moriya, K.: Japan's Historical Place Names 27: Place Names of Kyoto. Heibonsha Publishing (1979) (in Japanese)
16. Sakai, M., Yamada, S., Onoda, T.: Initialization of k-means method using independent component analysis. In: The 24th Annual Meeting of the Japanese Society for Artificial Intelligence (2010) (in Japanese)
17. Hyohanki Reading Circle, The Index of Hyohanki's Persons' Names. Shibunkaku Publishing (2007) (in Japanese)
18. Kurahashi, K., Mori, F.: Clustering with K-means algorithm by division and merging. Report of the Institute of Electronics, Information and Communication Engineers (IEICE), PRMU-106(470), 67–71 (2007) (in Japanese)

Author Index

Athanasiou, Spiros 124

Ballatore, Andrea 78
Banaei-Kashani, Farnoush 50
Bertolotto, Michela 78

Carswell, James D. 203
Claramunt, Christophe 5, 94
Coast, Steve 4

Efentakis, Alexandros 124

Galiero, Raffaele 109
Giorio, Clemente 109
Golay, François 188

Hattori, Fumio 36

Ingensand, Jens 188
Itsubo, Sho 230
Iwasaki, Hirotoshi 21

Jaramillo, Andres 94

Kimura, Fuminori 230
Kitayama, Daisuke 142
Kobayashi, Kaori 142
Kruijff, Ernst 159

Lerin, Pablo Martinez 176
Liagouris, John 124
Liang, Steve 64

Maeda, Akira 230
Martino, Sergio Di 109
Martins, Bruno 219
Mata, Felix 5, 94

Nurminen, Antti 159

Ohshima, Hiroaki 21
Oku, Kenta 36
Osaki, Takahiko 230
Oyama, Satoshi 21

Pfennigschmidt, Stefan 124
Pfoser, Dieter 124

Rezel, Rohana 64

Shahabi, Cyrus 50
Shirani-Mehr, Houtan 50
Sumiya, Kazutoshi 142

Takahashi, Naohisa 176
Takahashi, Yuku 21
Tanaka, Katsumi 21
Tezuka, Taro 230
Tsigka, Eleni 124

Veas, Eduardo 159
Voisard, Agnès 124

Xie, Xing 1

Yamamoto, Daisuke 176
Yamamoto, Mitsuo 21
Yin, Junjun 203

Zhang, Lin 50