Lecture Notes in Geoinformation and Cartography

Series Editors: William Cartwright, Georg Gartner, Liqiu Meng,
Michael P. Peterson

Liqiu Meng · Alexander Zipf · Stephan Winter
(Eds.)

Map-based Mobile Services

Design, Interaction and Usability

Springer

Editors

Prof. Dr. Liqiu Meng
Technische Universität München
Lehrstuhl für Kartographie
Arcisstr. 21
80333 München
Germany
meng@bv.tum.de

Prof. Dr. Alexander Zipf
Arbeitsgruppe Kartographie
Das geographische Institut
Universität Bonn
Meckenheimer Allee 172
53115 Bonn
Germany
zipf@geographie.uni-bonn.de

Dr. Stephan Winter
Senior Lecturer
Department of Geomatics
The University of Melbourne
Victoria 3010
Australia
winter@unimelb.edu.au

ISBN: 978-3-540-37109-0 e-ISBN: 978-3-540-37110-6

Lecture Notes in Geoinformation and Cartography ISSN: 1863-2246

Library of Congress Control Number: 2008921353

Cover design: deblik, Berlin

Printed on acid-free paper

9 8 7 6 5 4 3 2 1

springer.com

Preface

The worldwide popularisation of mobile communication technologies and the increasing awareness of usability issues since 1990's have been urging map designers to specialise and extend cartographic semiotics, visualisation styles and map use techniques for mobile contexts and small display devices. As a follow-up to the first book "Map-based Mobile Services – Theories, Methods and Implementations" published in 2005, this new one is devoted to design strategies, user interactions and usability issues. It addresses methods and techniques for topics that range from design and rendering, context modelling, personalisation, multimodal interaction to usability test. Instead of striving for a seamless coverage of all essential theoretical and technical issues with an equal depth and extent, we attempt to pinpoint a number of research highlights and representative development activities at universities, research institutions and software industry. The operational prototypes and platforms reported in the book are on the one hand outcome and feasibility proof of various approaches. On the other hand, they serve as a new starting point for the refinement of user interfaces and iterative usability tests.

The book is intended not only for cartographers, surveying engineers and geo-information scientists engaged in the development of location-based services, but also for software engineers and cognitive scientists working with interface design and usability assessment. In addition, we try to provide a number of real-life case studies for students, academics and practitioners from GIS, computer graphics and other relevant disciplines.

We gratefully acknowledge the authors of individual chapters for their generous contribution to this book project. Thanks are due to our peer reviewers for their constructive critics and suggestions. Finally, we would like to express our sincere appreciation to Mrs. A. Fleißner and Mr. H. Fan at the Department of Cartography, Technical University of Munich, for their technical assistance.

Liqiu Meng
Alexander Zipf
Stephan Winter

Table of Contents

Part I: Design Strategies and Rendering Techniques

Part III: Usability and Applications

1 The State of the Art of Map-Based Mobile Services

Liqiu MENG

Department of Cartography, Technical University of Munich, Germany

Abstract. This chapter raises a number of research questions with respect to design issues, interactions and user modelling of map-based mobile services. It gives an overview of the most common mobile user tasks, dominating presentation styles on mobile devices as well as their adaptation forms. The emphasis is laid on usability issues. Depending on whether a usability test is conducted before, during or after the design process, different strategies of user modelling can be applied. The general methods of constructing and evaluating usability tests are also described. Finally, the structure of the book with three theme blocks is outlined.

1.1 Research questions and development paradigms

Map-based mobile services are cartographic presentations on small display devices intended for interactive use in mobile environments. They represent one of the fundamental and most widespread types of location-based services (LBS). On the one hand, the technical restrictions of mobile devices and the dynamic usage contexts need to be considered as design constraints. On the other hand, the availability of additional situative information within the mobile environment offers new ways for data integration and individualisation of mobile maps. Although it has been generally agreed upon the fact that the immediate comprehensibility and the intuitive user interfaces are indispensable for the acceptance of mobile map services, various research questions concerning interaction paradigms and usability issues of cartographic presentations are still open.

The generation of context-aware mobile maps is mainly related to the following questions:

- Which user tasks are typically related to mobile maps?
- What kinds of design patterns from conventional mapmaking are re-usable?
- Which map contents and presentation styles are relevant for which communication goals and which situative context?
- Which generalization operations are meaningful for mobile maps?
- How far can the relevant context factors be categorised and formalised?
- How much detail on the map is necessary for an adequate interpretation of the user?
- How do situative context and user-specific parameters mutually affect each other?
- Which design strategies are possible to direct the attention of the user towards the focus (regarding content and location) of the map?

With regard to interactions, the designer of mobile services would typically ask:

- What intuitive interaction mechanisms and modalities (language, sound, keyboard, mouse, gestures, augmented reality etc.) in which combination are good for mobile maps?
- What impact does the choice of the technical interface (PDA, mobile phone, TabletPC, head-mounted device etc.) have on the selection of information and its map format? What are the ergonomic characteristics of these different devices?
- Which and how multimedia and hypermedia elements can be meaningfully integrated in mobile maps?
- When are 3D visualization and other map-like presentations meaningful and helpful on mobile devices?

Those who are engaged with user modelling tasks have to answer the questions like:

- What are the expectations regarding the appearance of maps related to socio-demographic characteristics or cultural background of the users?
- What characteristics are suitable indicators for preferences and abilities of a mobile user?
- How can user characteristics be inferred and weighted in a non-intrusive manner?
- What differences of user behaviour and memory load can be caused by the mobile usage environment in comparison to the stationary usage environment?

Finally, there are a number of methodological concerns such as

- How can cartographic rules and their relative priority be determined for various kinds of individualisation tasks?
- What experiments are suitable for the acquisition of user information and how can they be designed?
- What computing measures can be defined and determined?

Being constrained by evolving technologies and marketing strategies, the still very short history of map-based mobile services can be characterised by three development paradigms that are being concurrently practised: (1) designer-orientation since the beginning of 1990ties; (2) activity-orientation since mid 1990ties, and (3) usability-orientation since the turn of the new millennium. The first paradigm typically took the form of push-service under a threefold assumption that the mobile user has a potential demand on map services, the accessibility of operational map services would invoke or strengthen the user demand, and a meaningful mobile map for the developer would also make sense for users. The one-way nature, hence relative blindness of advertising push-services, however, was soon overwhelmed by the rapidly growing knowledge about the mobile activities. With the increasing awareness of the fact that many mobile users would be more interested in those map services that could essentially support their activities at hand in a mobile context, the design process became more focused on the activity-driven user requirements. The resulting maps began to take the character of double-way "pull"-services, aiming at providing mobile users with the right information at the right place and right time. Nevertheless, pull-services would be hardly accepted if they are not able to respond in real time to user's changing expectations that rely on and change with the dynamic mobile environments. Without sufficient knowledge about the interplay between the mobile user and his environment, a pull-service might be delivered inadequately or even at a wrong moment. The third paradigm, therefore, tries to look more at the overall mobile usage environment where additional up-to-date information and/or computing devices are accessible and can be collaboratively used with mobile maps. Based on the context-aware analysis, more rational requirements and quality measures for the design of mobile maps can be derived.

1.2 An overview of existing mobile map services

In a mobile environment, the user has two fundamental actions: (a) move from one place to another, and (b) stay where he is and looks around. In order to perform a concrete mobile task, he usually has to repeat these two actions several times and chain them in a reasonable sequence.

Today's map-based services typically support the following mobile tasks:

- Find the actual user location,
- Find locations of objects or people relevant to the current user,
- Plan a route,
- Guide a city tour,
- Navigate and orientate for different movement modes such as walking, cycling and driving,
- Retrieve information of landmarks,
- Simulate traffic noise, emergency, disasters etc., and
- Support the fleet management.

Being mainly developer-oriented and action-driven, the currently available map-based services can be categorised as follows:

(a) Mobility support

  - "*You-will-go*" service – One or many optimal routes between two given points are calculated and visually highlighted on the basis of available traffic information and various criteria such as the speed, distance, security and sightseeing (Radoczky and Gartner, 2005).
  - "*You-are-here*" service – The map graphics is dynamically adapted so that the actual location of the user is always visible (Sayda et al., 2002).
  - "*Find-next*" service – The map graphics is dynamically adapted so that both the next destination and the actual location of the user are visible (Klippel, 2003).
  - "*Way-finding*" service – The route with starting, intermediate and terminating stations and necessary landmarks in the surroundings is visualised at a map scale or LoD suitable to the movement mode (Bieber, 2004; Kolbe, 2004).
  - "*City guide*" service – Scenic spots selected by the user are visualised in multimedia (Paelke et al., 2005). The vicinity area is displayed at a higher LoD than the peripheral area (Etz & Haist, 2005).

(b) Information acquisition

- "*Event calendar*" service – A number of location- and time-relevant events such as conferences, exhibitions etc. within a user-defined area are classified and visualised (Hampe et al., 2005).
- "*Tour suggestion*" service – Tours that consider the personal preferences (sport, recreation, etc.) are displayed along with routing instructions (Holweg, 2004).
- "*Land mark*" service – The semantic information specifying individual land marks or their higher LoDs are displayed or hidden upon user request (Elias et al., 2005).

(c) Information communication

- "*Group diary*" service – Members of a mobile group inform each other of their actual locations with a sketch. Every informed member may modify or enrich the sketch with new information and distribute it to other group members. In this way, the sketch is shared by the group as a common memory (Kopczynski & Sester, 2004; Schulz, 2005).
- "*Group activity*" service – The map graphics is dynamically adapted so that different locations of group members are kept visible. Depending on the movement mode, the orientation of the map either remains constant or is dynamically adapted to the viewing direction or moving direction of each individual group member (Cheverst et al., 2000).

1.3 Adaptations and interactions

Usually a mobile user has a time-critical task at hand. Therefore, he would expect from the mobile device a personalised and non-intrusively rendered service that supports instead of distracting him. For this reason, the ideal mobile geo-services should possess the largest adaptability and require the least interactivity. Currently available mobile maps allow the following two adaptation forms:

- With help of sensory techniques – The mobile device acquires the actual location, moving direction and moving speed of the user by means of a GPS receiver, a digital compass or other accessible sensors in the environment. At the same time, the map graphics is automatically refreshed so that the user could always get a personalised or egocentric presentation with a number of selected landmarks in his actual vision field (Frank et al., 2004).
- With help of user inputs – The system prompts the user to input some of his personal data such as age group, preferred language, movement mode etc. The presentation style is then automatically tailored to fit this particular user or user group (Sarjakoski & Sarjakoski, 2004; Reichenbacher, 2004).

In practice, both forms can be integrated in one system which allows the adaptation to be driven by a combination of dynamic situative parameters with some static user parameters.

The interaction between the user and a mobile map normally takes place in either a monomodal manner (e.g. touch pen) or a multimodal manner (e.g. combination of touch pen, speech and gesture). The following operations with a mobile map are possible:

- Panning – The user may stepwise move the map towards different directions.
- Zooming – The user may enlarge or reduce the display window without content changes.
- Zooming with LoD – When the user enlarges or reduces the display window, a new level of detail will be rendered which is either pre-calculated or generated in real time.
- Hiding and revealing – The user may visually hide or highlight certain objects or object classes.
- Switching – The user may choose different complementary presentation styles.
- Focusing – The user may click at a certain object and retrieve its detailed information stored in a database.
- Tuning of visualisation parameters – The user may change his viewing angle and graphic variables in terms of colour, texture, symbol size, and figure-ground contrast etc. within the allowed value ranges.
- Dialogue – The user may activate a dialogue window and input his personal data.
- Query – The user may search for certain objects or object classes by giving one or many criteria.

1.4 The usability of mobile map services

The development of usable mobile map services faces three research challenges: (1) identify the relevant information, (2) transmit it in real time, and (3) render it in an immediately comprehensible form. Challenge (1) and (2) are relatively straightforward due to their strong dependence on user activities or technical possibilities, while Challenge (3) requires extensive and precise knowledge about the cognitive processes and memory capacities of mobile users in situ.

Similar to the evaluation of a general design product, the usability of a mobile map service can be measured in three aspects - effectiveness, efficiency and user satisfaction (Dickmann, 2005; Meng, 2005; Sarjakoski & Nivala, 2005). So far, the majority of usability tests have been focused on the determination of effectiveness that reflects the functionality of a service and efficiency that deals with the performance as well as the cost-benefit issue. There is a growing awareness of the fact that the acceptance degree of a service on the market depends additionally on the degree of user satisfaction. Since this latter aspect is rather subjective and related to user's emotional state, it remains a bottleneck for the usability researchers to find a generally agreeable and repeatable measurement.

Usability investigations of mobile map services with subjects in simulated or real usage context have the main objective to detect the map use performance and its correlations with user properties. A usability test can be intrusively or non-intrusively conducted before, during or after the development of mobile map service. During an intrusive test, the tester may interrupt or "bother" the subject from time to time, for example, by asking him to have a dialogue or explain his behaviour of performing a certain task. A non-intrusive test takes place without any interference of the tester. Usually, the user behaviour is observed and recorded by sensors such as video camera, eye tracking devices, fMRI (functional magnetic resonance imaging) etc.

The user information captured in various ways during various stages of service design will be then statistically analysed, which leads to the identification of representative map use problems, a categorisation of user stereotypes or parameterised user model. The insight gained from usability tests will support the service designer to infer user requirements and determine suitable design rules or patterns.

1.4.1 Pre-design usability test

A usability test before the design practice is so to speak disconnected from the service to be designed. It is driven by the belief that without seeing what an ideal mobile map service would look like, the subjects can enjoy their neutrality and freedom of imagination, although their experiences have large impacts on their imaginations. Questionnaires, interviews, scenarios and controlled experiments are prevalent methods applied in a pre-design usability test.

Questionnaires are used to capture the relevant demographic attributes and prior knowledge of subjects concerned with hand-held devices and mobile map use. Interviews allow the designer and the user to experience a common cognitive "walk-through" that helps specify the information demand and alternative design solutions for the given mobile tasks. Scenarios describe retrospectively the social environment and the personal behaviour of performing mobile tasks, especially the encountered problems. Information such as personal constructs, spatial capabilities, selected activities and critical events can be derived from scenarios. Finally, some general ideas, opinions to different visualisation styles can be collected by means of controlled experiments such as sketches, mock-ups and presentations designed for other usage contexts.

Wealands et al. (2005) reported a two-stage investigation. In the first stage, user attributes were captured and categorised by means of questionnaires. It resulted in a rough user profile composed of three aspects – user properties, usage context and user preferences. In the second stage, subjects were observed and interviewed in order to document the personal problems in handling with mobile tasks. The evaluation of this additional information led to a refined user model.

van Elzakker (2004) designed an experiment which allowed subjects with different experiences and demographic properties to describe their desired visualisation services for the given applications, sketch their personal design solution with help of graphic drawing tools, and finally give reasons for their solution based on "Thinking-aloud" principle. Such an experiment yielded a matching matrix between different applications and various design solutions. In addition, the behaviour difference between novice users and expert users was documented.

A large number of comparative studies between different design styles for desktop and mobile context have revealed that the multimedia products such as animations, travel simulations and virtual fly-throughs generally allow fast object recognition (Tversky et al., 2002; Shelton & McNamara, 2004; Hakala et al., 2005; Harrower & Sheesley, 2005; Cartwright, 2006). However, they are not superior to 2D abstract maps in terms of giving an overview and spatial relations. The subjects tend to be cognitively overloaded without having control over the visual and temporal properties of the presented scene (Fuhrmann, 2003).

1.4.2 Participatory usability test

A usability test during the design process of a mobile map service, also termed as participatory usability test, serves the main purpose to discover usability problems or sources of irritation with a mobile map being designed (Rosson & Carroll 2002). A mobile map service would not be usable, if one of the design steps does not meet user's expectations. The problems from early design steps, if not solved in time, may accumulate and propagate to later steps. Theoretically, the overall usability measure can be defined as a weighted value of many component measures corresponding to different steps of the workflow ranging from selection, encoding, decoding to memorising of the information (Swienty, 2005). Ideally, a usability test should be conducted for every step along the workflow so that the discrepancy between user's cognitive capability and the affordance of the service could be more precisely analysed.

It is particularly meaningful to embed as many interactive operations as possible in early steps of the design practice. This would give the subjects a sufficient freedom within the allowed scope to personally determine the suitable data amount for a given mobile task, adjust the design parameters and interaction modalities. Being documented in a log-file, these interactions contain important clues about difficulties of map use, preferences for certain interactive operations and correlations between user stereotypes and their preferred design styles.

Winter & Tomko (2004) believed that a mobile map that takes the bodily experience of its user into account could reduce the cognitive workload of reading. Based on their observations of the postures of mobile map users, they proposed to shift the conventionally centred user location to the bottom of a map so that the mobile user has access to the information from a larger vision field in his viewing direction. This approach sounds reasonable because the lower border of the display lies near to the body and the mobile user primarily perceives the surrounding in his viewing direction.

Through extensive experiments, Hermann & Gibbert (2003) captured ergonomic properties of mobiles map users for navigation tasks. They developed a design pattern language for user interface of mobile mapping systems, with the intension to apply it as a tool to support similar mobile tasks. A further design concept can be found in Burghardt et al. (2005) according to which the frequently called operations are made more accessible than other operations.

1.4.3 Post-design usability test

A post-design usability test is conducted after a prototype of mobile map service has been completed. Many reported experiments have a holistic nature which attempt to test whether or how far the map service can convey its pre-determined affordance to the target user (van Elzakker, 2005). The qualitative and quantitative usability statements about a prototype play a decisive role for the verification or improvement of the design rules. However, they seldom lead to entirely new design solutions. In case that the prototype fails to work as expected, it is not possible to precisely identify the causes.

Questionnaires, behaviour observation, thinking-aloud and controlled experiments are typical methods applied in a post-design usability test. User behaviour is mainly observed non-intrusively with video recording, registration of click actions and eye movement registration. Thinking-aloud method can be applied during or after the behaviour observation with the aim to document personal impressions and reasons for the interactions. The controlled experiments capture the effectiveness and efficiency of the prototype which are usually characterised by error rate and speed of performing given tasks.

Radoczky & Gartner (2005) conducted a user test for their pedestrian navigation systems and confirmed the hypothesis that a schematic presentation (topogram) combined with selected landmarks as well as text from a city plan is more suitable than topogram alone (too little information) or city plan alone (too much information). Reichenbacher & Abel (2005) confirmed the belief that routing maps in combination with 3D perspectives and meaningful language instructions work well with the built-in device in car. A direct transferring of this combination for outdoor applications, however, is not straightforward due to further constraints. In another similar user test, Chittaro & Burigat (2005) reported that the arrow symbol in a topogram or in front of an image background was a helpful orientation support for most users.

The user test from Wakabayashi (2005) showed that the navigation performance was more strongly influenced by the visualisation service than by the spatial experiences of users, especially when complex routes were involved. Sarjakoski and Nivala (2005) tested their adaptive geovisualisation approach for mobile information acquisition with experts and real users. They found out that the usability of their approach was significantly influenced by age difference and culture difference.

1.4.4 Methods for the construction and evaluation of usability tests

The majority of the reported usability investigations of mobile maps so far are targeted to derive and justify the qualitative or quantitative statements about the potential of a given mobile map service for typical usage scenarios and user groups. If the objective is to identify usability problems, a small test group with less than 10 subjects would be sufficient. Empirically, 80% of usability problems can be detected with only 4 to 5 users (Virizi, 1992). However, if a usability test aims at confirming design hypotheses which require statistically significant evidence, a much larger test group (e.g. with >50 subjects) is necessary. Due to the limited practicability of large-scale investigations, it is difficult to recruit and simultaneously involve a large group of subjects in the usability test. Often, the subjects are tested in small groups and at different times with slightly different test conditions. The possible fluctuations in the results should be minimised by introducing a number of correction parameters.

The development of formal and extensive usability tests for mobile map services are still in its beginning stage. Theoretically, the construction of usability test and the selection of subjects need to consider three general quality criteria according to (Lienert & Raatz 1998) – validity, reliability and objectivity. The evaluation of user data is based on the probabilistic theory and the theory of measurement error.

User information captured through questionnaires, interview and scenarios reveals a varying objectivity. The answers with relatively low objectivity can only be intuitively evaluated based on the insight and knowledge of testers. If, however, the user gives answers following a pre-defined standard, a formal categorisation is possible. User statements which reflect the personal constructs allow an evaluation based on some rational criteria. Video records, eye movement tracks or brain activities exhibit a high objectivity. Their evaluation can be combined with the "thinking-aloud" approach. On the one hand, a "top-down" strategy can be applied to scrutinise how far the user behaviour coincides with expectations of the designer. On the other hand, a "bottom-up" strategy is useful to infer from the registered behaviour information, the visual impacts of various graphic variables.

1.5 About the book

As a follow-up of the first book published in 2005, the individual chapters of this new one are contributed by invited specialists and selected participants of the second workshop on "Map-based mobile services" 2005 in Salzburg, Austria. The support of authors from nine different countries has made it possible for us to give an overview of high-end development activities at universities, research institutions and software industry. The book contents are grouped into three parts.

The four chapters in Part I are dedicated to design strategies and rendering techniques for mobile devices. *Burigat* and *Chittaro* present an algorithm of icon aggregation constrained by legibility requirements. *Elias* and *Paelke* introduce a number of user-centred design concepts and their implementations for building landmarks of different categories. A strategy of incremental transmission for fast access to multi-resolution data in client-server architecture is described by *Follin* and *Bouju*. Finally, *Plesa* and *Cartwright* demonstrate a "proof-of-concept" prototype of a non-realistic 3D map for mobile devices.

The six chapters in Part II deal with issues of context modelling, personalisation of mobile maps and user interactions. *Palmonari* and *Bandini* introduce a method of commonsense spatial reasoning with the objective to correlate heterogeneous information sources acquired from distributed devices. An approach for the implicit and explicit acquisition of user information and user-oriented content providing is developed and implemented by *Weakliam et al*. *Doyle et al.* conduct a comparative study of multimodal interfaces for mobile mapping systems with the emphasis on the combination of speech and gesture input. For a routing query task, *Jokinen* investigates user preferences concerned with speech and pen pointing gesture and the correlation between the two interaction modes. *Nurminen* and *Oulasvirta* demonstrate their design ideas in a prototype system that supports spatial updating and alignment of physical and virtual spaces.

The five chapters in Part III are focused on usability issues and various mobile applications of map services. A window-based routing algorithm with mobile maps is introduced and evaluated by *Karimi* and *Sutovsky*. *Hochmair* conducts a usability study to explore the preferred interaction modes between the user and a PDA device in case of signal dropouts during the indoor-navigation. *Reilly et al.* examine the effect of combining mobile maps with other navigation aids for spatial knowledge acquisition in a mobile context. *Beeharee* uses map information as the basis for other mobile services in the real world such as visibility computation of real life entities and de-cluttering treatment constrained by screen size of mobile devices. Based on an analysis of mobile location-based gaming systems, *Paelke et al.* give an overview of the application potential of maps for game design. Finally, *Rinner* introduces a method of multi-criteria decision analysis which can provide personalised decision support for emergency response.

1.6 Concluding remarks

The current development is characterised by two complementary trends. On the one hand, many researchers have been actively experimenting with high-end techniques such as XML (*Extensible Markup Language*) and XSLT (*Extensible Stylesheet Language Transformation*) for multipurpose publication of mobile maps, XML-based SVG (*Scalable Vector Graphics*) and Macromedia Flash for the generation of interactive or animated maps on mobile devices. Many other researchers, on the other hand, have been intensively investigating the transferability and extensibility of traditional design theories to mobile usage context. Design solutions such as focus maps, relevance-driven symbolisation, egocentric map, map gestures and a number of design patterns for recurring mobile tasks are suggested and implemented (Zipf & Richter, 2002; Meng, 2005; Reichenbacher, 2005). Such a double-tracked development has led to an improved insight into the nature of mobile map services. Unlike stationary maps which are primarily designed to communicate the descriptive information about where, what, how much etc. for interactive use in a relatively placid mood, mobile maps are usually consulted in a hasty mood, therefore, they should not only contain the right information amount that fits the capacity of the short-term memory, but also minimise the cognitive effort by directly guiding users how to do and in which sequence.

References

Bieber, G. (2004): Intelligente Tourenplanung für Fußgänger. *INI - GraphicsNet 2004: Verkehr und Mobilitätssicherung, Impressum*, pp.36-39.

Burghardt, D., Edardes, A., Purves, R. and Weibel, R. (2005): Automatische Generalisierung dynamisch generierter Karten für mobile Endgeräte. *Kartographische Nachrichten* 3/2005, pp.119-125.

Cartwright, W. (2006): Using non-immersive multimedia to build surrogate travel tools. *Kartographische Nachrichten* 3/2006, pp.131-136.

Cheverst, K., Davies, N., Mitchell, K., Friday, A. and Efstratiou, C. (2000): Developing a Context-aware Electronic Tourist Guide: Some Issues and Experiences. *Proc. of CHI 2000,* ACM. Hague, the Netherlands, pp. 17-24.

Chittaro, L. and Burigat, S. (2005): Augmenting audio messages with visual directions in mobile guides: an evaluation of three approaches. *Mobile HCI 05*, pp. 107-114.

Elias, B., Hampe, M. and Sester, M. (2005): Adaptive Visualisation of Landmarks using an MRDB. In: *Meng, L., Zipf, A. & Reichenbacher, T. (eds): Map-based Mobile Services*. Springer, pp. 73-86.

Elzakker, C.P.J.M. van (2004): The use of maps in the exploration of geographic data. Utrecht / Enschede: KNAG / Faculteit Geowetenschappen Universiteit Utrecht / ITC. Netherlands Geographical Studies 326.

Elzakker, C.P.J.M. van (2005): From map use research to usability research in geo-information processing. *Proc. of International Cartographic Conference* 2005 (CD-ROM), A Coruña, Spain.

Etz, M. and Haist, J. (2005): Mobile 3D viewer. In: Computer Graphic - User-centered visualization and interface design, 4/2005, Vol.17, pp. 28.

Fuhrmann, S. (2003): Supporting Wayfinding in Desktop Geovirtual Environments. In Peterson, M. (ed.): Maps and the Internet. Amsterdam, Cambridge. Elsevier Press.

Hakala, T., Lehikoninen, J. and Aaltonen, A. (2005): Spatial interactive visualization on small screen. Mobile HCI 05, pp. 137-144

Harrower, M. and Sheesley, B. (2005): Moving beyond novelty: Creating effective 3D fly over maps. *Proc. of International Cartographic Conference* 2005 (CD-ROM), A Coruña, Spain

Hampe, M., Hose, K., Hatger, C. and Katterfeld, C. (2005): A web based and time dependent visualisation of an event calendar. *Proc. of International Cartographic Conference* 2005 (CD-ROM), A Coruña, Spain.

Hermann, F. and Gibbert, R. (2003): UI design patterns für Navigationssysteme auf mobilen Geräten. *1. Workshop des German Chapters der Usability Professionals Association e.V.* pp. 199-203.

Holweg, D. (2004): Mobile Fußgängerinformationssysteme. *INI - GraphicsNet 2004 Verkehr und Mobilitätssicherung*, Impressum. pp. 24-27.

Klippel, A. (2003): Wayfinding Choremes: Conceptualizing Wayfinding and Route Direction Elements. PhD thesis. Department of Mathematics and Informatics, University of Bremen.

Kolbe, T. (2004): Augmented panorama and videos for pedestrian navigation. In: *Proc. 2nd Symposium on Location Based Services and TeleCartography 2004, Geowissenschaftliche Mitteilungen,* TU Wien.

Kopczynski, M. and Sester, M. (2004): Representation of Sketch Data for Localisation in Large DataSets, In: *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*.

Lienert, G.-A. and Raatz, U. (1998): Testaufbau und Testanalyse. 6. Auflage. Beltz, PsychologieVerlagsUnion. Chap. 10-12.

Meng, L. (2005): Egocentric Design of Map-Based Mobile Services. *The Cartographic Journal*, Vol. 42, No.1, pp.5-13.

Meng, L. (2006): Mobile Geovisualisierungsdienste und ihre Gebrauchstauglichkeit. *Wissenschaftliche Arbeiten der Fachrichtung Geodäsie und Geoinformatik der Leibniz Universität Hannover*, Nr. 263, Festschrift, 125 Jahre Geodäsie und Geoinformatik, Hannover. 249-2006.

Paelke, V., Elias, B. and Hampe, M. (2005): The CityInfo Pedestrian Information System - An Experiment with Content Creation and Presentation Techniques. *Proc. International Cartographic Conference* (CD-ROM), A Coruña, Spain.

Radoczky, V. and Gartner, G. (2005): Extent and effectiveness of map abstraction for communicating routes in a LBS. *Proc. International Cartographic Conference* (CD-ROM), A Coruña, Spain.

Reichenbacher, T. (2004): Mobile cartography - Adaptive visualisation of geographic information on mobile devices. PhD thesis. Technical University of Munich.

Reichenbacher, T. (2005): The Importance of Being Relevant. *Proc. International Cartographic Conference* 2005 (CD-ROM), A Coruña, Spain.

Reichenbacher, T. and Abel, R. (2005): Visualisierung in Navigationssystemen und mobilen Geodiensten. *Kartographisches Taschenbuch* 2005, Kirschbaum, pp. 29-50.

Rosson, M.B. and Carroll J.M. (2002): Usability Engineering – Scenario-based development of human-computer interaction. *Morgan Kaufmann Publishers,* Chap. 7.

Sarjakoski, L.T. and Nivala, A.-M. (2005): Adaptation to Context - A Way to Improve the Usability of Mobile Maps. In: *Meng, L., Zipf, A. & Reichenbacher, T. (eds): Map-based Mobile Services*. Springer, pp. 107-123.

Sarjakoski, L.T. and Sarjakoski, T. (2004): A use case based mobile GI service with embedded map generalisation. ICA Workshop on Generalisation and Multiple Representation, August 20-21, 2004, Leicester, CD-ROM.

Sayda, F., Reinhardt, W. and Wittmann, E. (2002): Positionsbezogene Dienste zur Unterstützung von Bergsteigern und Wanderern. In: *A. Zipf / J. Strobl (Eds.): Geoinformation mobil 2002.* Wichmann, pp. 127-137.

Shelton, A. L. and McNamara, T. P. (2004): Orientation and Perspective Dependence in Route and Survey Learning. *Journal of Experimental Psychology: Learning, Memory, and Cognition* 30 (1): pp. 158-70

Schulz, T. (2005): Graphic- and sketch-based interaction for GIS applications. *Computer Graphik Topics – Reports of the INI-GraphicsNet* 1/2005 Vol.17, pp. 13-14

Swienty, O. (2005): The Development of Explorative Geovisualisation. *Proc. of International Cartographic Conference* 2005 (CD-ROM), A Coruña, Spain.

Tversky, B., Morrison, J. B. and Betrancourt, M. (2002): Animation: can it facilitate? *International Journal of Human-Computer Studies*, 57, pp. 247-262

Virizi, R. A., 1992: Refining the test phase of usability evaluation: How many subjects are enough? *Human factors*, Vol. 34, No. 4, pp. 457-468.

Wakabayashi, Y. (2005): Necessary conditions for cartographic communication and navigation with guide maps. *Proc. of International Cartographic Conference* (CD-ROM), A Coruña, Spain.

Wealands, K., Cartwright, W., Miller, S. and Benda, P. (2005): User assessment for developing optimal cartographic representation models within an Australian mobile location-based services travel. *Proc. of International Cartographic Conference* 2005 (CD-ROM), A Coruña, Spain.

Winter, S. and Tomko, M. (2004): Shifting the Focus in Mobile Maps. *ICA Workshop UPIMap2004*, Tokyo.

Zipf, A. and Richter, K.-F. (2002): Using Focus Maps to Ease Map Reading. *Künstliche Intelligenz* (4), pp. 35-37.

2 Decluttering of Icons Based on Aggregation in Mobile Maps

Stefano BURIGAT, Luca CHITTARO

HCI Lab, Department of Math and Computer Science, University of Udine, Italy

Abstract. In this chapter, we will deal with cluttering issues associated with the visualization of a high number of icons, representing Points of Interest, on mobile maps. Clutter is a frequently occurring problem that makes it difficult to understand the visualization and has the even more critical effect of masking data. As the user zooms out, the positions of icons increasingly converge until they partially or entirely overlap each other. In regions of high icon density, this effect impairs even the effectiveness of a close-up view. Finding ways to visualize icons in a comprehensible uncluttered fashion is a challenging task. We will first discuss techniques which have been previously proposed in the literature for the automatic placement of labels on maps, a problem which is closely related to the placement of icons. We will then propose an approach for icons - aggregation - aimed at mitigating cluttering issues when icon density is high. The approach is based on an aggregation algorithm to reduce the total number of icons that need to be displayed and on the design of icons that could function as both individuals and aggregates.

2.1 Introduction

In recent years, a number of map-based applications and services have been made available to users of mobile devices, with a particular emphasis on Personal Digital Assistants (PDAs). In this context, maps often need to be adapted to specific user requirements which are not known a priori. Therefore, one cannot pre-compute and store all possible maps that may be needed by users but they have to be generated in real-time. This is a particularly challenging task that requires, for example, automatic placement of text labels and icons which are essential parts of any map. Unfortunately, the design of map-based applications is constrained by device limitations: techniques and practices that are effective in the desktop scenario cannot be simply adapted to mobile devices but must be redesigned to achieve usability and performance goals on such equipment and novel solutions are often needed to cope with specific issues (Chittaro, 2006).

In this chapter, we will deal with the specific problem of visualizing a large number of icons, each one representing a Point of Interest (POI) on mobile maps. The visualization of many icons on the same screen often leads to cluttering issues, especially when users perform zoom-out operations and icons begin to touch and overlap each other. This may degrade the effectiveness of even a close-up view of a map and mask other important map features such as roads. For example, displaying the

Fig. 2.1. An example of icon cluttering taken from the Yahoo Maps (maps.yahoo.com) site: the displayed icons represent restaurants in the Manhattan area. Many icons overlap and have masked almost all other map features of an area.

location of restaurants as icons on a map might work very well at a large scale, but it could become useless at smaller scales when most icons overlap, as illustrated in Fig. 2.1. Typically, some simple criteria are used to limit the number of icons to be displayed. A widespread but extreme solution lies in displaying only those POIs selected by the user from a list. A slightly more flexible solution subdivides the list of all POIs (ordered by name or other criteria) in pages and displays only one page at a time (consisting of a fixed and usually limited number of POIs). However, these techniques are only suitable for the simultaneous display of map contents and a (mostly) textual list of POIs. They are hardly applicable to the mobile context characterized by limited screen space. Moreover, reducing the number of POIs to be displayed may not solve all overlapping issues, as illustrated in Fig. 2.2.

Finding ways to visualize icons in an easy-to-understand uncluttered fashion, while satisfying various users' requirements is a challenging task. The limitations of mobile devices further exacerbate the problem. For example, since only limited screen space is available, the amount of icon clutter on overviews increases with respect to desktop screens, and because of the limited processing power, the use of powerful but computationally intensive algorithms is discouraged.

While many papers address problems pertaining to the design of icons or the encoding of values with icons (Wong and Bergeron, 1997), the problem of how to properly place icons on maps avoiding clutter has been scarcely investigated. A possible strategy to tackle the problem is to adapt methods and algorithms that have been developed for the closely related problem of label placement on maps. In the first part of this chapter, we will thus describe label placement techniques which have been previously proposed in the literature, mainly for desktop map-based applications, and which are of interest for fast icon placement. We will then propose techniques based

Fig. 2.2. An example of icon cluttering with a limited number of icons, taken from the Expedia (www.expedia.com) site.

on an approach – aggregation – that aims at handling high icon density. The approach is based on an icon aggregation algorithm to reduce the total number of icons that need to be displayed and on the design of icons that could function both as individuals and aggregates in such a way that they can convey critical information on the associated POIs and possibly provide access to less critical information by tapping on them. We will conclude with a discussion of future research directions on icon decluttering techniques for mobile map-based applications.

2.2 Label placement in map-based applications

Placing text labels on maps while avoiding overlaps with cartographic symbols and other labels is a fundamental problem in the field of cartography. Label placement is still often performed manually (Cook and Jones, 1990), although several techniques have been proposed to automate it.

Usually, there are three different label placement tasks: labeling of area features (e.g., countries, lakes, oceans), line features (e.g., roads, rivers), and point features (e.g., cities on small-scale maps, mountain peaks, points of interest). When multiple features have to be handled, these tasks share a common combinatorial difficulty in automatic label placement.

Most techniques for automatic label placement presented in the literature focus on point features. To formulate the point-feature label placement (PFLP) problem as a combinatorial optimization problem, it is necessary to define a *search space* and an *objective function*.

An element of the search space (called *labeling*) is a mapping of point features into label positions, where the potential label positions for a feature are usually taken from an explicitly enumerated set. A typical set of four possible label positions for a point feature is shown in Fig. 2.3.

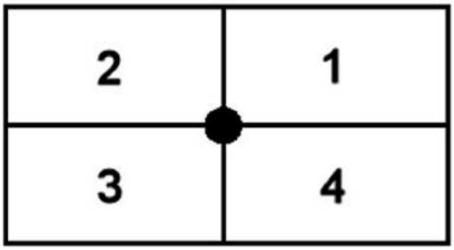


Fig. 2.3. A set of potential label positions for a feature located in the center (black dot). Each rectangle corresponds to a region where the label may be placed and the values indicate the desirability of each position (the lower the value, the more preferred the position).

The objective function assigns to each labeling a score that corresponds to the quality of that labeling. Labeling quality depends on many factors, including map purpose and characteristics of human visual perception. A commonly used objective function for the PFLP problem was proposed by Yoeli (1972). In Yoeli's proposal, the quality of a labeling depends on the following criteria:

- The amount of overlap between labels and graphical features (including other labels);
- Cartographic preferences among potential label positions (a canonical set is shown in Fig. 2.3);
- The number of point features that are left unlabeled.

The objective function specifies how to compute a numerical score from these criteria. Lower scores are usually assigned to better labelings, so that the goal of the search is to minimize the objective function.

Unfortunately, while some simple restrictions of the label placement problem can be easily solved in polynomial time (Formann and Wagner, 1991), the general search problem and many of its interesting variants are Non-deterministic Polynomial-time hard (NP-hard) (Marks and Shieber, 1991). Therefore, a complete search algorithm could be impractical and a practical algorithm incomplete. Research efforts have then been made to find heuristic methods that work acceptably in practice, although they may not exhibit guaranteed performance.

In the following sections, we will briefly describe some of the approaches to label placement proposed in the literature. We will also introduce an established data structure, the *conflict graph*, which supports label placement and plays a fundamental role in our icon aggregation approach.

2.2.1 PFLP algorithms

A basic approach to PFLP is to perform an *exhaustive search* of the solution space. The algorithms based on this approach process points in a prescribed order, trying to place each label in a position that is currently unobstructed. When it is impossible to label a point, either because there are no positions without conflict or because all available positions have been tried, these algorithms backtrack to the most recently labeled point and consider the next available position. Exhaustive search algorithms stop when an acceptable labeling has been identified or when the entire search space has been explored. Various heuristics, such as *variable ordering*, *value ordering*, *source-of-failure* and *pruning*, can be employed to improve the algorithm performance (Korf, 1988). Variable ordering consists in processing the most difficult points first, that is, those points which have fewer labeling options available. Value ordering is based on a prioritization of potential label positions for each point so that later conflicts can be avoided. A common method to achieve this goal is to always choose the label position which has the smallest number of conflicts with other features or labels. The source-of-failure heuristic backtracks to one of the points conflicting with the currently considered point, instead of backtracking to the most recently placed point. Pruning heuristics are used to eliminate whole areas of the search space from consideration. For example, it is possible to avoid considering label positions which conflict with a large number of surrounding positions. Unfortunately, regardless of these heuristics, exhaustive search is impractical as a general solution to the PFLP problem because the search space has an intrinsic exponential nature.

More efficient algorithms are based on avoiding the backtracking strategy of exhaustive search, thus limiting the scope of the search. Unlike exhaustive search algorithms, these *greedy algorithms* (Christensen et al., 1995) may leave unlabeled any point whose label cannot be placed, or may allow a certain degree of overlap among labels, thus providing a tradeoff between labeling quality and computational cost. Like exhaustive search algorithms, greedy algorithms exploit various heuristics, such as those previously described, to guide the search and help identify reasonable labelings.

The quality of labelings produced by a greedy algorithm can be considerably improved if labelings are subsequently repaired by local alteration. *Gradient-descent algorithms* (Hirsch, 1982; Christensen et al., 1995) can be used for this purpose. Starting from an initial labeling, usually obtained by placing labels randomly in any of the available candidate positions, the basic idea of gradient descent is to iteratively implement the label repositioning by moving in a descending direction towards a minimum of the objective function.

The main weakness of gradient-descent algorithms is their inability to escape from local minima of the objective function. Stochastic methods such as *simulated annealing* (Kirkpatrick et al., 1983) attempt to solve this issue by incorporating a probabilistic or stochastic element into the search, allowing movement in different directions from that of the gradient, thus including the possibility to temporarily worsen the objective function.

Various other approaches, such as *integer programming* (Zoraster, 1990), *genetic algorithms* (Verner et al., 1997) and *tabu search* (Yamamoto et al., 2002), have been proposed in the literature but their detailed analysis is beyond the scope of this chapter. For an extensive list of references on map labeling, we refer the reader to the Map-Labeling Bibliography web site (Wolff, 2006).

2.2.2 The conflict graph

The conflict graph is a structure that describes conflicts due to overlaps between label positions or, more accurately, between the rectangles defining label positions (see Fig. 2.3). Each node of the conflict graph corresponds to a label position while the edges of the graph link together conflicting label positions. More formally, a conflict graph can be defined as an undirected graph G = (N, E), where N is a list of nodes and E is a list of edges. The degree of a node is the number of incident edges for that node and is a measure of the amount of conflict of the corresponding label position with other label positions. The higher the degree of a node, the more difficult it is to place the associated label on a map.

The conflict graph can be represented as an adjacency matrix, where rows and columns represent nodes and each entry is either 1 if the corresponding nodes are connected or 0 if they are not connected. In an adjacency matrix, the degree of each node is the sum of values in a line or in a column. An alternative representation of the conflict graph is an adjacency list, where each node is associated to the list of nodes it is connected to.

To build the conflict graph, it is necessary to check for overlaps among the considered label positions. A naive approach to perform this test is the all-pairs algorithm where each label position is tested against all other ones. In spite of its simple principle, this approach has a quadratic cost in the number of label positions and, therefore, is too time-consuming and inappropriate except for very small sets of labels. A more efficient algorithm is based on the sweep-line approach (de Berg et al., 1997). In general, a sweep-line algorithm detects all intersections among a set of axis-parallel rectangles in the plane by making an imaginary vertical line sweep across the plane from left to right. The algorithm is supported by two data structures, called *event-point schedule* and *sweep-line status*. Event points are the x-values where the sweep line must stop because either its status changes or intersections have to be reported. In the case of labels, event points are the x-values of the left and right edges of the rectangles defining label positions and the event-point schedule is the sorted list of these points. The sweep-line status stores intervals corresponding to the intersections of the sweep line with the given rectangles. The endpoints of the intervals are the y-values of the lower and upper edges of the input rectangles. Initially, the sweep-line status is empty. When a left edge of a rectangle is encountered during the sweep, the interval corresponding to the edge is inserted into the sweep-line status. A rectangle is reported if its interval is currently in the sweep-line status and intersects the new interval. When a right edge of a rectangle is encountered, the corresponding interval is deleted from the sweep-line status. The label overlap problem is thus reduced to the problem of maintaining a set of intervals in such a way that they can be efficiently inserted and deleted, and interval-intersection queries can be answered quickly. An interval-tree data structure (Edelsbrunner, 1980) can be employed to achieve this result. The computational complexity of the sweep-line algorithm is $O(NlogN + I)$ where N is the number of rectangles and I is the number of intersections.

The conflict graph can be used to reformulate the map labeling problem as a maximum independent set problem (Strijk et al., 2000). In particular, finding a solution for the map labeling problem is equivalent to finding a subset S of the vertices of the graph with maximum size, for which there is no edge (u,v) in the graph that connects vertices u and v in S. Moreover, cost values can be assigned to the vertices to model cartographic preferences for the candidate label positions and the sum of the costs of the vertices in S is minimized with respect to the size of S. The vertices in S thus indicate the candidate positions that must be used for labeling the points and, if none of the vertices related to a particular point appears in S, then the point remains unlabeled.

Various algorithms have been proposed to solve the labeling problem as a maximum independent set problem. For example, Kakoulis and Tollis (1998) propose a general approach to labeling based on the conflict graph and its connected components. They use a greedy heuristic for maximum independent set to split these components into cliques (a clique in an undirected graph G, is a set of vertices V such that for every two vertices in V, there exists an edge connecting the two). Finally they construct a graph where the identified cliques and the features of the problem (points, lines or area) become nodes. In this graph, a feature and a clique are joined by an edge if the clique contains a candidate of the feature. A maximum-cardinality matching is then used to obtain the final labeling. Wagner et al. (2001) present a fast, simple and effective algorithm that can be applied to point, line, or area labeling. The input to the algorithm is the conflict graph of the label candidates. The algorithm is organized into two phases. In the first phase, a set of rules is used to simplify the conflict graph without reducing the size of an optimal solution. The goal of this phase is to reduce as much as possible the number of label candidates of the features. In the second phase, a heuristic is employed to reduce the number of label candidates of each feature to at most one. The heuristic is conceptually simple and consists in eliminating the most troublesome candidates first, by going through all features that still have the maximum number of candidates and deleting the candidate with the maximum number of conflicts. This process is repeated until each feature has at most one candidate left and no two candidates intersect.

Conflict graphs can also be pre-computed to improve the performance of label placement algorithms when real-time generation of maps is necessary (e.g., in mobile or web-based interactive applications). Indeed, in map-based applications, different sets of labels are usually presented to the user at different zoom levels. For example, an overview of a city map might contain only labels for the most important roads and locations while more detailed views will show also labels of less important features. Usually, the set of labels varies only at predefined zoom levels, called scale levels, while it does not change at intermediate levels. As a consequence, panning and zooming operations that do not change the scale level will not change the amount of label overlaps, since no labels will be added or removed from the considered map. Therefore, it is possible to build conflict graphs (one for each scale level) in a preprocessing phase, thus improving the global performance of the map labeling process (Yamamoto et al., 2005). Pre-computation of the conflict graph is an acceptable approach for mobile map-based applications where the number of possible scale levels is fixed. A more flexible approach is presented by Petzold et al. (2004), who propose the *reactive conflict graph*, a data structure that is built in a pre-processing phase and that models possible label conflicts regardless of map scale. In particular, specific algorithms are used to identify the scale ranges at which potential conflicts among features may occur and these ranges are assigned to the edges of the graph. The reactive conflict graph is then stored in a three dimensional geometric data structure, e.g. the

3-D-R-Tree (Guttman, 1984), where the first two dimensions represent features geometry (or their bounding boxes) and the third dimension represents map scale. During labeling, the reactive conflict graph is queried to obtain a static conflict graph that contains all objects to be labeled and all possible conflicts among these objects for a specific map area and scale.

2.3 Decluttering of icons through aggregation

While, as shown in previous sections, a considerable effort has been devoted to automating the placement of labels on maps, much less attention has been devoted to the automatic placement of icons. This may be due to the consideration that icon placement can be regarded as a special case of label placement. Indeed, if we treat icons as labels, it is possible to use PFLP algorithms for automatic icon placement on maps. For example, Harrie et al. (2005) discuss a method of combining label and icon placement on maps created in real-time. Their method is based on reducing the search space (by choosing a fixed number of possible placements for each object) before performing a combinatorial optimization step based on simulated annealing to find a possible solution. The advantage of reducing the search space is that it allows to decrease the processing time (or improve the solution within a limited number of iterations), even if it may cause the optimal solution to be missed.

However, there are some subtle differences between the label placement and icon placement problems that can make it inappropriate to directly apply known PFLP algorithms to icons. First, labels can be removed from a map if, at the end of the labeling process, they cannot be placed without conflicts. This approach cannot be applied to icons because their purpose, in most cases, is to point out the existence of specific objects (POIs) that would be otherwise impossible to detect on a map. Their removal would thus cause fundamental information to be missing. Second, unlike labels, POI icons usually occupy a predefined fixed position with respect to their corresponding map feature, generally being placed exactly over the feature location. This constraint may be actually considered an advantage from the point of view of the computational cost of the placement algorithms. Indeed, a larger set of icon positions would result in a problem with a larger number of possible solutions and, in spite of the possibly better-looking results, the growth in problem complexity would result in a much higher computational effort. However, it must be noted that, despite the reduction in complexity, the icon placement problem still remains NP-hard and heuristic algorithms are thus needed to solve it within reasonable time.

In the following sections, we will describe *icon aggregation*, the approach we propose to tackle the icon placement problem while taking into consideration its peculiarities. The idea is to identify clusters of mutually overlapping icons and replace them with special aggregator icons, whose purpose consists in pointing out the presence of aggregated icons as well as providing users with a means to access information about each of them. Indeed, selecting an aggregator icon produces a pop-up window that lists the aggregated POIs and allows the user to obtain more information on each POI. This technique can thus help to declutter the visualization, freeing map space by removing all icon clusters without causing information to be permanently lost.

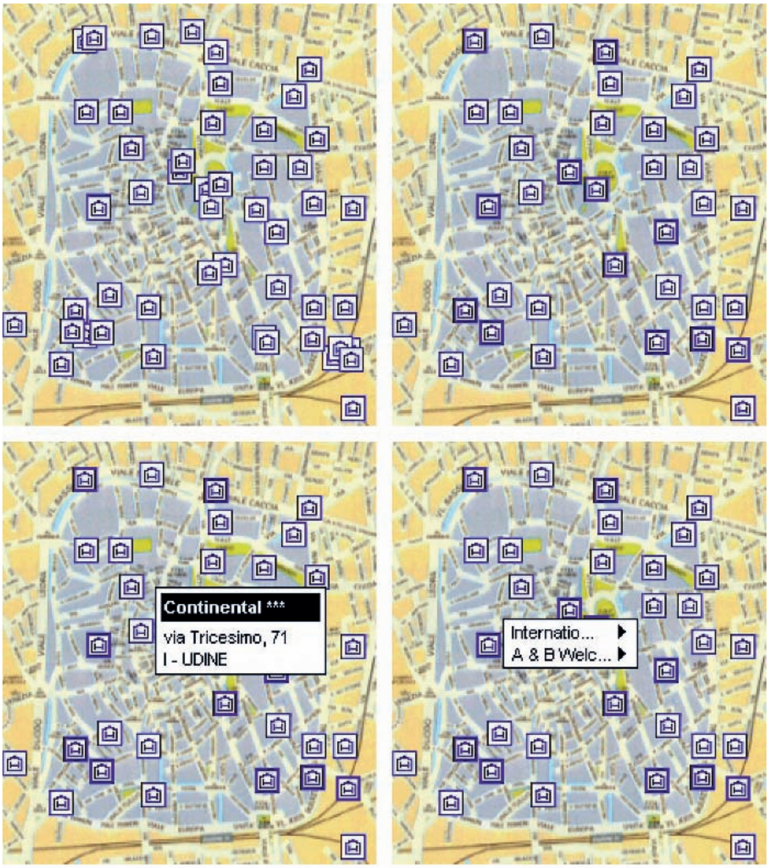


Fig. 2.4. An example of icon aggregation: all clusters of overlapping icons that can be seen in the original map visualization in the upper left image are replaced by special icons representing aggregations (as displayed in the upper right image). Selecting an individual icon allows one to obtain detailed information on the corresponding POI, while selecting an aggregator icon produces the list of aggregated POIs.

An example of aggregation is illustrated in Fig. 2.4: the image on the upper left displays a map with 50 icons, some of which overlap; the image on the upper right displays the same map after aggregation. As shown, clusters of overlapping icons have been removed and replaced with aggregator icons that are graphically different from those used for individual POIs (in this case, hotels). As illustrated in the bottom left image, selecting an individual icon allows one to obtain detailed information on the corresponding hotel (in this case, its name, number of stars and address). Selecting an aggregator icon instead produces the list of aggregated hotels, as illustrated in the bottom right image, and each item in the list allows one to access detailed information on the corresponding hotel. It must be noted that aggregation may also highlight the presence of icons that are almost completely hidden by others, as in the case of the

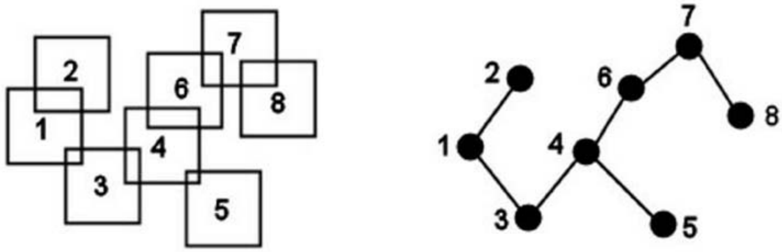


Fig. 2.5. A set of overlapping icons (left) and the corresponding conflict graph (right).

aggregator icon to the left of the pop-up window in the bottom right image. In the original map, the fact that an icon almost completely overlapped another icon there could pass unnoticed.

Although the idea behind aggregation is simple, different issues must be addressed to implement the approach. The most important issue consists in designing appropriate algorithms to identify clusters of icons and process them. Given the limitations of mobile devices, we need algorithms that seek a compromise between cost and benefit, with good quality (in terms of number of placed icons) and a short response time. Indeed, since we do not actually lose data when performing aggregation, we may want to sacrifice optimality of the final solution in return for improved performance. In this way, it becomes possible to update the visualization in real-time, as users perform operations (e.g., panning) that may change the set of icons to visualize.

Our implementation divides the task in two distinct subtasks that can be treated separately. The first subtask is to identify overlaps between icons and build a static conflict graph to store this information. As an example, Fig. 2.5 shows a set of overlapping icons and the corresponding conflict graph. We used the algorithm presented in section 2.2.2 to implement the conflict graph as an adjacency list. The second subtask is to process the information provided by the conflict graph so that a proper aggregation of the icons can be determined. We implemented different algorithms to perform this subtask to identify the one providing the best trade-off between performance and output quality.

2.3.1 Icon aggregation algorithms

The conflict graph described in the previous section is the input for the algorithm that determines how to aggregate icons. Because of the limitations in terms of processing power of the target platform on which aggregation has to take place, we chose to favor performance over quality, thus implementing only greedy but fast algorithms that could provide acceptable results.

Given a conflict graph, the *maximum aggregation algorithm* we propose produces a set *S* of aggregated icons without conflicts. The algorithm works as follows:

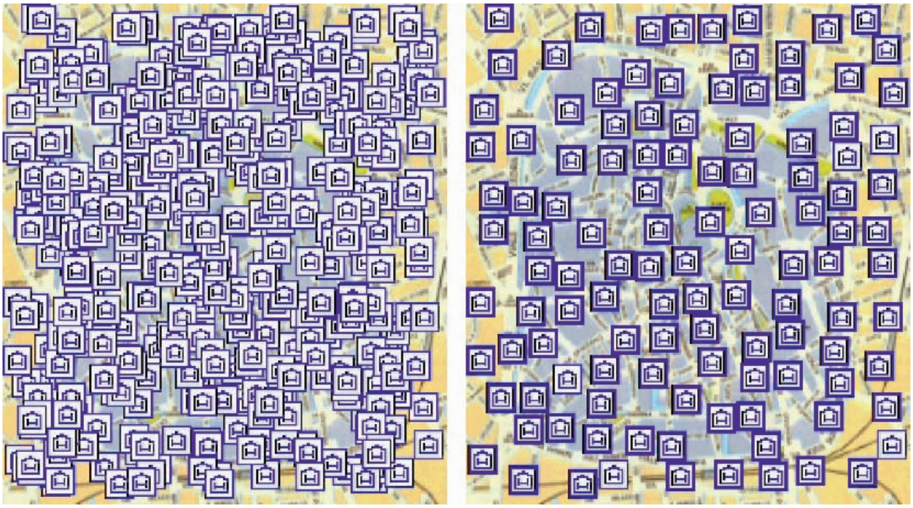


Fig. 2.6. An example of icon aggregation with the maximum aggregation algorithm on a map with 400 icons: the original map is displayed on the left, the result of the aggregation is displayed on the right.

- Initially, the solution set *S* is empty and the active node set *ANS* is equal to the full node set *N*.
- If *ANS* is empty, exit with *S* as the result. Otherwise, repeat the following steps:

1. Calculate the degree of all nodes in *ANS*.
2. Select n_{min}, the node with the smallest degree in *ANS*.
3. If the degree of n_{min} is higher than *0*, then mark n_{min} as aggregator and mark all nodes adjacent to n_{min} as aggregated to n_{min}.
4. Place n_{min} in the solution set *S*.
5. Remove n_{min} and all nodes adjacent to it from *ANS*.

An implementation of this algorithm builds a priority queue from the conflict graph, ordering icons in ascending order of degree, where the degree of each icon is the number of conflicts for that icon. At each step, the algorithm takes the icon with the smallest degree, removes it from the priority queue and places it in the solution set. The icon is marked as aggregator if its degree is higher than *0*, i.e., if it overlaps other icons in the queue. Moreover, all icons that are in conflict with the aggregator are removed from the queue, they are marked as aggregated and the degrees of their neighbors that are still in the queue are decreased accordingly. The process is then repeated until the priority queue is empty.

Fig. 2.6 shows an example of the output of this algorithm on an icon placement problem involving 400 icons.

The algorithm is simple but provides good solutions in a time that depends on how the priority queue is implemented. This is because the algorithm spends most of its time adding elements to the queue, removing elements from the queue or changing the priority of elements. In particular, updating the degree of the neighbors of all nodes adjacent to n_{min} (which is needed after having removed these nodes from *ANS*) requires an update of the priority queue and is quite time-demanding.

To implement the priority queue, we first tried the skip list data structure (Pugh, 1990). Skip lists are a probabilistic alternative to balanced trees (the most common data structure for implementing priority queues) that allows operations such as insertion and deletion of elements to be much simpler and faster. Unfortunately, in all our practical tests involving up to 400 icons, the cost associated to the management of a priority queue based on a skip list was higher than using a plainly ordered list. We thus employed simple lists for the final implementation of the algorithm.

An interesting effect of using the maximum aggregation algorithm is that, by considering icons in increasing order of degree, the number of icons associated to each aggregator tends to be balanced. Indeed, when icons with an initially high number of conflicts are actually processed, it is highly probable that their degree has decreased because of previous icon removals from the priority queue. This is desirable because aggregations will be composed of an approximately similar (and limited) number of elements, and users will appreciate this feature when examining the results on a map.

To further improve the performance of the aggregation process in terms of processing time, we looked for a way to simplify the maximum aggregation algorithm. In particular, since updating the priority queue after the degree of its elements has changed is time-consuming, we tried to avoid this operation entirely. The resulting *fast aggregation algorithm* goes through almost the same steps as the previous algorithm but does not need to recalculate the degree of all nodes in the active node set and to update the priority queue accordingly. By processing icons only in their initial ascending order of degree, this modified algorithm is faster than the original algorithm

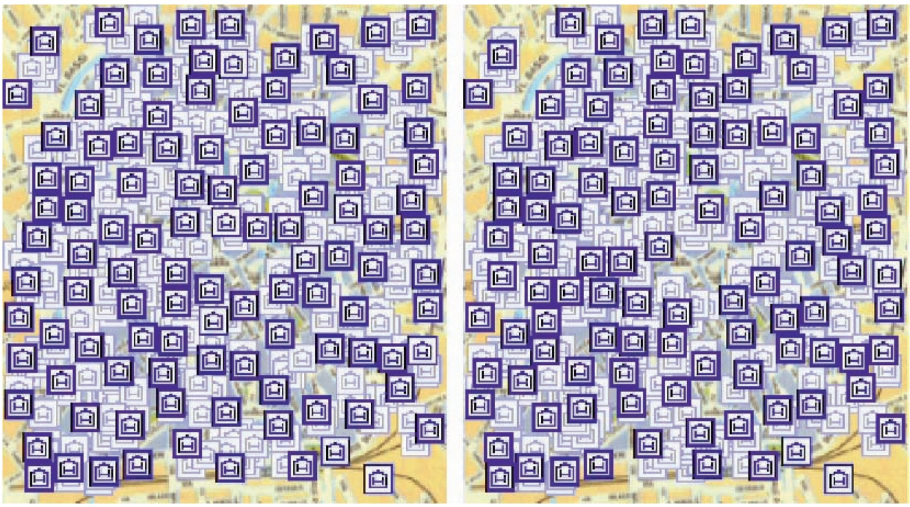


Fig.2.7. Output of the proposed algorithms for icon aggregation on a set of 400 randomly generated icons which have been left (shaded) in the background for reference purposes. The image on the left refers to the fast aggregation algorithm and the image on the right to the maximum aggregation algorithm. Despite the differences in the solution sets, the two images contain an approximately equal number of icons (108 vs. 109).

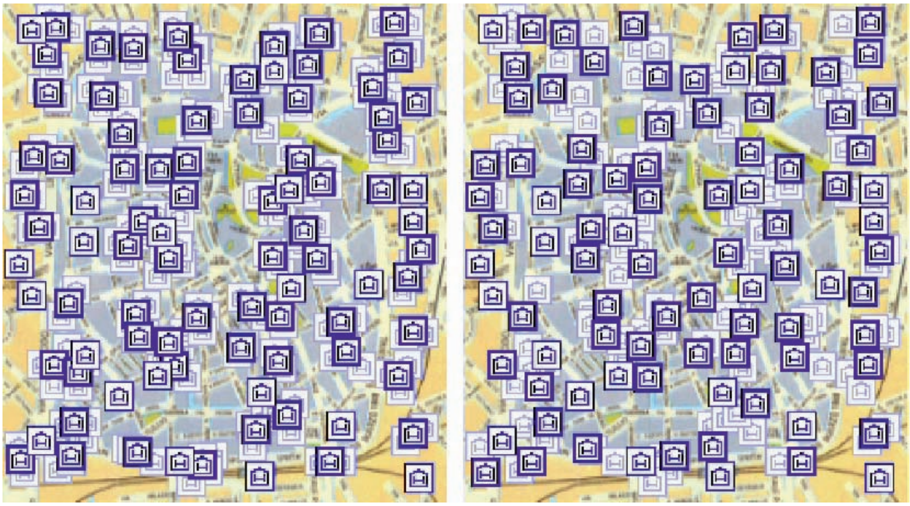


Fig. 2.8. Two different techniques to place aggregator icons: on the left, aggregators are placed in the centroid of each cluster of aggregated icons which have been left (shaded) in the background for reference purposes; on the right, aggregators are in the same position as one of the icons in each cluster. As it can be seen, aggregators may overlap each other with the first technique.

while achieving similar results in terms of the number of displayed icons. Fig. 2.7 shows a comparison between the results of the two algorithms on a set of 400 randomly generated icons.

A feature of these algorithms that we did not discuss so far is the placement of aggregator icons. In the current implementation, aggregator icons are placed exactly at the same position of one of the icons they aggregate. An alternative technique consists in placing aggregator icons in the centroid of each cluster of icons to be aggregated. The advantage of such a solution is that, from a geometric point of view, each aggregator is in a more representative position since its Euclidean distance from any of the aggregated icons positions is at most $(a*\sqrt{2})/2$, where a is the size of the icon, while in the current implementation this distance can reach $(a*\sqrt{2})$. Unfortunately, this technique suffers from a disadvantage that substantially reduces its usefulness and has prevented it from being adopted. Indeed, there is no guarantee that aggregator icons will not overlap each other, as illustrated in Fig. 2.8.

2.3.2 Relaxing the overlap constraint

In most map-based applications, the purpose of icons is simply to point out the location of specific geographic objects. Even when there are clusters of overlapping icons, there is often enough visual information for the user to be able to distinguish individual icons and determine their location on the map. However, the algorithms presented so far try to remove all overlaps among icons. While this is a fundamental requirement

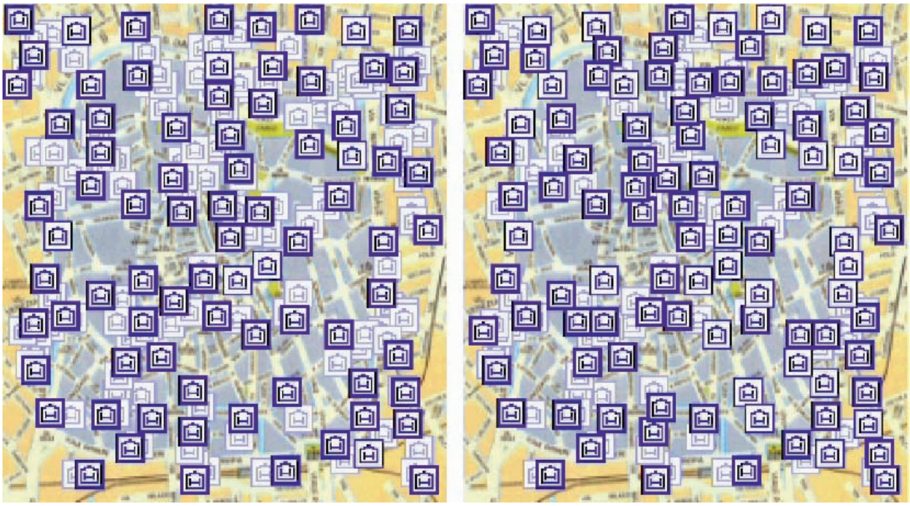


Fig. 2.9. Effect of relaxing the overlap constraint on icons: 20 more icons have been placed on the map on the right (where a certain degree of overlap is tolerated) with respect to the map on the left (where all overlaps have been removed).

for label placement, it might be considered too strict for icon placement. For example, looking at Fig. 2.4 it is possible to identify clusters of icons that severely overlap as well as icons that barely touch each other. In this latter case, it could be better to avoid aggregating the considered icons. In general, it would thus be useful to relax the overlap constraint, aggregating icons in such a way that a certain degree of overlap between icons is tolerated.

There are two possible ways of managing the relaxation of the overlap constraint. The simplest approach, which is the one we implemented, consists in building a custom version of the conflict graph where an arc between two nodes is added only when the degree of overlap between the corresponding icons exceeds a fixed threshold. Standard algorithms such as those presented in the previous sections could then be applied to perform the aggregation. A more flexible approach is based on associating a cost value with each arc in the conflict graph, corresponding to the amount of overlap between icons, and then modifying the aggregation algorithm so that this value is taken into consideration when processing icons. Fig. 2.9 shows how relaxing the overlap constraint leads to an increase in the number of icons that can be placed on a map when compared to the standard non-overlap approach.

2.3.3 Increasing map legibility

The algorithms presented in section 2.3.2 aim at maximizing the number of icons without conflicts displayed on a map, while keeping processing time short. By removing icons from the visualization, these algorithms also have the effect of enhancing the legibility of the map (i.e., the portion of map that is not hidden by icons), especially

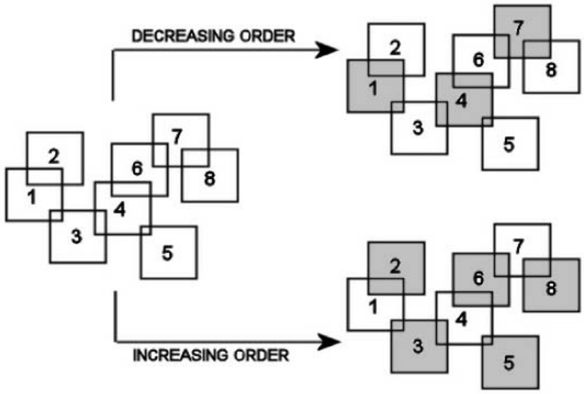


Fig. 2.10. Comparison of the outcome of the fast aggregation algorithm when processing nodes in decreasing or increasing order of degree. Icons in the solution set are highlighted in gray.

in highly cluttered regions, as illustrated in Fig. 2.6. However, it is possible to improve map legibility, which is a fundamental requirement in many map-based applications, by further reducing the number of icons displayed on the map. To achieve this goal, we tried two different approaches, producing two variants of the fast aggregation algorithm (similar variants for the maximum aggregation algorithm can also be produced). The two approaches can also be combined to provide better results.

The first variant is based on processing icons in decreasing (instead of increasing) order of degree, starting from icons with a higher number of overlaps. In this way, larger clusters of overlapping icons will be removed early in the process, possibly reducing the cardinality of the solution set. Fig. 2.10 shows a comparison of the outcome of the fast aggregation algorithm when processing icons in decreasing or increasing order of degree. In the considered example, when using decreasing order, the first icon that is processed is 4 (which becomes the aggregator for icons 3-4-5-6), followed by icon 1 (aggregator for 1 and 2) and icon 7 (aggregator for 7 and 8). When using increasing order, the first icon to be processed is 2 (aggregator for 1 and 2), followed by icon 5 (aggregator for 4 and 5), icon 8 (aggregator for 7 and 8), icon 3 (individual icon), and finally icon 6 (individual icon).

The second variant is based on removing all icons within a fixed distance K (in the conflict graph) from the currently processed icon. For example, for K = 2, the neighbors of an icon as well as the neighbors of those neighbors will be removed from the map and aggregated. Increasing the parameter K allows one to further reduce the number of visualized icons (until only one icon for each connected component in the conflict graph will remain) but has also the effect of increasing the maximum Euclidean distance between an aggregator and its aggregated icons.

Fig. 2.11 shows an example of the outcome of the two proposed variants compared with the original fast aggregation algorithm.

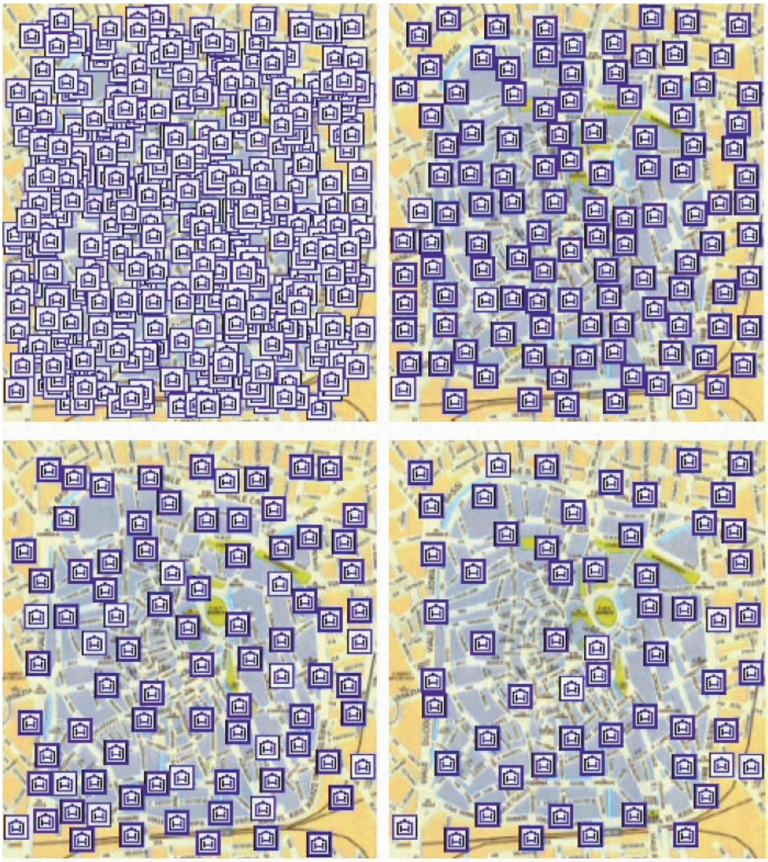


Fig. 2.11. Effect of different icon aggregation algorithms on map legibility. The original icon set is depicted in the upper left image, the other images show respectively the result of the fast aggregation algorithm (upper right) and its decreasing degree (lower left) and 2-distance (lower right) variants.

2.3.4 Evaluation

We implemented the algorithms presented in the previous sections using the C# language and the .NET Compact Framework for PocketPC devices and run them on a Dell Axim X30 device (equipped with a 624MHz processor) on different randomly generated sets of icons under the following conditions:

- Screen resolution: 240 x 268 pixels.
- Fixed icon size: 16 x 16 pixels.
- Icon set size: N = 50, 100, 200, 400.

- For each icon set size, 25 different trial configurations with random placement of icons.

Table 2.1 Mean number of icons displayed after aggregation.

Algorithm	50	100	200	400
Min-overlap	39.8	66.2	107.3	148.4
Maximum	36.9	58.2	87	116
Fast	36.9	58.2	86.6	111.1
Fast (inverse order variant)	34.3	50.5	67.1	86
Fast (2-distance variant)	34.1	48.8	60.4	67.8

Table 2.1 reports the mean number of icons displayed after aggregation for different icon set sizes. The results show that the maximum and fast aggregation algorithms (whose goal is to maximize the number of displayed icons) are almost equivalent with 50, 100 and 200 icons, while they slightly differ with 400 icons, with the first algorithm performing better than the second. We also included results for a variant of the maximum aggregation algorithm that allows a minimal overlap (3 pixels) among icons, thus being able to place more icons for each icon set size.

Both variants of the fast aggregation algorithm (whose goal is instead to increase map visibility) reduce the number of displayed icons compared to the original algorithm with a difference that starts to be significant for icon set sizes higher or equal than 100 icons. Moreover, the 2-distance variant ($K = 2$) performs better than the inverse order variant in highly cluttered configurations involving high numbers of icons.

Table 2.2 Mean computation times (in seconds) to obtain a solution on a Dell Axim X30.

Algorithm	50	100	200	400
Maximum (skip list)	0.043	0.177	0.502	1.848
Maximum (plain list)	0.017	0.040	0.109	0.380
Fast	0.014	0.032	0.077	0.194
Fast (inverse order variant)	0.014	0.031	0.077	0.185
Fast (2-distance variant)	0.011	0.022	0.048	0.142

Table 2.2 compares the mean computation times of the algorithms. We distinguished the implementation of the maximum aggregation algorithm using the skip list data structure from the one using the plain list data structure. The results show that the fast aggregation algorithm is faster than the maximum aggregation algorithm (with plain list) but the difference starts to be significant only with 400 icons. The implementation of the maximum aggregation algorithm using the skip list is instead the slowest regardless of the number of icons, taking much more time when more than 200 icons are considered. Of the two variants of the fast aggregation algorithm, the

first one is almost equivalent to the original algorithm, while the second one performs slightly better with each icon set size.

2.4 Future research directions

The aggregation approach we proposed in this chapter is based on the design of proper aggregator icons to indicate clusters of individual overlapping icons. The basic solution we employed to design such aggregators consists in slightly varying the graphical appearance of individual icons by increasing the thickness of their border. Different approaches can be explored as well. For example, it is possible to indicate the number of aggregated icons by (slightly) increasing the size of aggregators. This would help users to detect areas of a map where higher numbers of icons have been aggregated.

In some map-based applications, icons are not simple placeholders that point out the presence and location of specific objects in an area but can also be used to convey information about the features characterizing these objects. For example, icons representing hotels might visually provide users with information about hotel quality, range of prices, services or other attributes. It is evident that in this case there is an even greater need to avoid overlaps among icons so that all useful information can be easily obtained by users. However, if we applied the aggregation approach as described in section 2.3, most of the visual information concerning aggregated icons would be lost. It is thus necessary to couple aggregation with additional techniques that can solve this issue. A trivial, far from optimal, solution would be to provide access to this information only through pop-up windows, by simply interacting with aggregators. Other solutions may generate aggregator icons that provide overviews of the attribute values of aggregated objects. For example, the aggregator may show the maximum value of each attribute characterizing aggregated objects so that users can understand at a glance if it is worth interacting with the aggregator to obtain more detailed information.

Icons may also refer to entities such as states, districts or parcels, and encode information about the different parameters characterizing them. Since these areas vary strongly in both shape and size, the task of positioning icons is not trivial and may require different approaches compared to the standard icon placement problem. An example of such an approach is proposed by Fuchs and Schumann (2004), who combine a simplified displacement algorithm with a suitable *focus&context* technique that interactively solves the remaining spatial conflicts.

Another variant of the icon placement problem takes into consideration icons belonging to different categories of POIs (e.g., hotels and restaurants). Since applying a standard aggregation algorithm to each set of icons separately does not guarantee that there will be no overlaps in the final outcome, proper solutions are needed to address this issue.

2.5 Conclusions

The requirements for icon placement on maps, although not trivial, have been scarcely studied by the research community. A number of methods for automatic label placement have been instead proposed in cartography. Unfortunately, they are focused on high-quality results, leading to long response times that are not suited for devices with limited computational resources and for interactive environments. In this chapter, we presented an approach to the icon placement problem that is based on aggregating overlapping icons, replacing them with special aggregator icons that point out the presence of aggregated icons and allow the user to retrieve information about each of them. We designed and evaluated fast algorithms to perform the aggregation and studied interesting variants of the problem.

It must be pointed out that none of the proposed algorithms can be an optimal solution to the icon placement problem for all possible map-based applications and their use scenarios. The most flexible approach, then, would be to choose the best solution according to users' needs and to the number of icons to manage. For example, when only a small number of icons have to be placed on a map, it could be better to relax the overlap constraint to maximize the amount of icons displayed. On the contrary, when a high number of icons must be considered, it could be better to adopt more aggressive aggregation algorithms, such as those presented in section 2.3.4, so that an adequate portion of map can still be visible. Thus, integrating different algorithms and selecting the most appropriate one when needed would improve the usefulness of any mobile map presented to the user.

Acknowledgements

This work has been partially supported by the Italian Ministry of Education, University and Research (MIUR) under the PRIN 2005 Project "Adaptive, Context-aware, Multimedia Guides on Mobile Devices".

References

Christensen, J., Marks, J., and Shieber, S. (1995): An empirical study of algorithms for point-feature label placement. *ACM Transactions on Graphics*, Vol. 14 No. 3, pp. 203-232.

Chittaro, L. (2006): Visualizing Information on Mobile Devices. *IEEE Computer*, Vol. 39 No. 3, pp. 40-45.

Cook, A.C., and Jones, C.B. (1990): A Prolog rule-based system for cartographic name placement. *Computer Graphics Forum*, Vol. 9, pp. 109-126.

de Berg, M., van Kreveld, M., Overmars, M., and Schwarzkopf, O. (1997): *Computational Geometry Algorithms and Applications*. Springer-Verlag, Berlin.

Edelsbrunner, H. (1980): Dynamic Rectangle Intersection Searching. Institute for Information Processing Report 47, Technical University of Graz, Austria.

Formann, M., and Wagner, F. (1991): A packing problem with applications to lettering of maps. In *Proc. Symposium on Computational Geometry*, ACM Press, pp. 281-288.

Fuchs, G., and Schumann, H. (2004): Intelligent Icon Positioning for Interactive Map-based Information Systems. In: *Proc. International Conference of the Information Resources Management Association* (IRMA 2004), New Orleans, USA.

Guttman, A. (1984): R-trees: A Dynamic Index Structure for Spatial Searching. In: *Proc Conference on Management of Data*, University of California, Berkeley, USA, pp 47-57.

Harrie, L., Zhang, Q., and Ringberg, P. (2005): A case study of combined text and icon placement. In *Proc. International Cartographic Conference* (ICC 2005, CD-ROM), A Coruña, Spain.

Hirsch, S. A. (1982): An algorithm for automatic name placement around point data. *American Cartographer*, Vol. 9 No. 1, pp. 5-17.

Kakoulis, K.G., and Tollis, I.G. (1998): A unified approach to labeling graphical features. In *Proc. Symposium on Computational Geometry*, ACM Press, New York, pp. 347-356.

Kirkpatrick, S., Gelatt Jr., C.D., and Vecchi, M.P. (1983): Optimization by simulated annealing. *Science*, Vol. 220, pp. 671-680.

Korf, R.E. (1988): Search: A survey of recent results. In: *Exploring Artificial Intelligence: Survey Talks from the National Conferences on Artificial Intelligence*, Shrobe, H.E. (Ed). Morgan Kaufmann, pp. 197-237.

Marks, J., and Shieber, S. (1991): The Computational Complexity of Cartographic Label Placement, Technical Report TR-05-91, Center for Research in Computing Technology, Harvard University.

Petzold, I., Gröger, G., and Plümer, L. (2004): Modeling of Conflicts for Screen Map Labeling In *Proc. International Society for Photogrammetry and Remote Sensing Congress* (ISPRS 2004, CD-ROM), Istanbul, Turkey.

Pugh, W. (1990): Skip lists: A probabilistic alternative to balanced trees. *Communications of the ACM*, Vol. 33 No. 6, pp. 668-676.

Strijk, T., Verweij, B., and Aardal, K. (2000): Algorithms for maximum independent set applied to map labelling. Technical Report UU-CS-2000-22, Department of Computer Science, Utrecht University.

Verner, O.V., Wainwright, R.L., and Schoenefeld, D.A. (1997): Placing text labels on maps and diagrams using genetic algorithms with masking. *INFORMS Journal on Computing*, Vol. 9 No. 3, pp. 266-275.

Wagner, F., Wolff, A., Kapoor, V., and Strijk, T. (2001): Three Rules Suffice for Good Label Placement. *Algorithmica*, Vol. 30 No. 2, pp. 334-349.

Wolff, A. (2006): Map-Labeling Bibliography. http://i11www.iti.uni-karlsruhe.de/~awolff/map-labeling/bibliography/

Wong, P.C., and Bergeron, R.D. (1997): 30 Years of Multidimensional Multivariate Visualization. In *Scientific Visualization*, Nielson, G.M., Hagen, H., and Muller, H., (Eds). IEEE Computer Society Press, pp. 3-33.

Yamamoto, M., Camara, G., and Lorena, L.A.N. (2002): Tabu search heuristic for point-feature cartographic label placement. *GeoInformatica*, Vol. 6 No. 1, pp. 77-90.

Yamamoto, M., Camara, G., and Lorena, L.A.N. (2005): Fast point-feature label placement for real time screen maps. In *Proc. Symposium on GeoInformatics* (GEOINFO 2005), Campos do Jordão, Brazil.

Yoeli, P. (1972): The logic of automated map lettering. *The Cartographic Journal*, Vol. 9 No. 2, pp. 99-108.

Zoraster, S. (1990): The solution of large 0-1 integer programming problems encountered in automated cartography. *Operations Research*, Vol. 38 No. 5, pp. 752-759.

3 User-Centered Design of Landmark Visualizations

Birgit ELIAS, Volker PAELKE

Institute of Cartography and Geoinformatics, Leibniz University of Hannover

Abstract. Landmark-based navigation is the most natural concept for humans to navigate themselves through their environment. It is therefore desirable to incorporate this concept into car and personal navigation systems. However, today's navigation systems are limited to driving assistance and provide guidance information in terms of instructions and distances, based on the current position and the underlying digital map. Research in the field of spatial cognition has shown that the use of landmarks is very important for humans navigating through unfamiliar environments. The integration of landmarks could therefore make navigation instructions more usable. In this chapter we present a design concept for the visualization of building landmarks in mobile maps. We consider four categories of building landmarks: well-known shops (trade chains), shops referenced by their type, buildings with a specific name or function and buildings described by characteristic visual aspects. We then examine how landmarks from each of these categories can be effectively visualized by comparing possible visualizations at different abstraction levels, ranging from photo realistic image presentations, over drawings, sketches and icons to abstract symbols and words. As a guideline to designers we provide a matrix representation of the design space from which possible and recommended presentation styles for each category can be identified.

3.1 Introduction

Maps are a very important means to provide spatial knowledge and communicate route information (MacEachren, 1995; Kray et al., 2003). Current pedestrian navigation systems rely heavily on maps in addition to positioning and routing functionality to convey wayfinding information to their users. Many recent research projects have developed prototypes for mobile services like GiMoDig, NEXUS, LoL@ and NAVIO. While some have focused on the technical aspects of mobile applications, others have examined the cartographic repercussions of small displays (Gartner and Uhlirz, 2001; Radoczky and Gartner, 2005). The effective integration of landmarks into such maps has not been examined in detail so far.

Research in the field of spatial cognition investigates the structure and elements of wayfinding instructions and provides another important foundation for the design of pedestrian navigation systems. Daniel and Denis (1998) have identified route actions (instructions about the next movement), orientations and landmarks as the basic components of (verbal) route directions. Further experiments have shown that the integration of landmarks into routing instructions enhances the perceived quality of the description (Denis et al., 1999). Tversky and Lee (1999) have compared the basic elements of route maps and route directions and found that both consist of the same underlying structure and semantic content.

Consequently, a good pedestrian route map should include the same elements as verbal directions. Landmarks will form an indispensable part of maps in mobile cartography applications and designers require appropriate visualization techniques for their effective presentation.

3.2 Related work

3.2.1 Landmarks in wayfinding instructions

Landmarks are significant physical, built or culturally defined objects that stand out from their surroundings and help in locating the geographic position (Golledge, 1999). They are classified as local and global or on-route and off-route landmarks (directly neighboured to the route or in the far distance like a tower or mountain chain). Furthermore, on-route landmarks are positioned between nodes, at decision points (a junction where a navigation decision is required) or at potential decision points (where a navigation decision is possible but the route goes straight on) (Lovelace et al., 1999).

Currently, landmarks are not yet part of commercial navigation data sets. In fact, all available route planning and guidance applications use data sets that are tailored to the requirements of car navigation. The upsurge of pedestrian navigation applications on mobile devices increases the demand to integrate important landmark information for pedestrians. If information about landmarks were available, it could be integrated into the database and used for wayfinding descriptions.

Various research approaches try to develop formal models or extract landmarks automatically from databases and focus on local landmarks at (potential) decision points (Raubal and Winter, 2001; Elias, 2003; Elias and Brenner, 2004; Elias 2006). These approaches are currently confined to the investigation of buildings as landmark objects. As a matter of fact, other topographic objects like parks, bridges, and railroad tracks are also suitable as landmarks and can be extracted from existing databases (Elias and Sester, 2002).

The integration of landmarks into wayfinding descriptions requires a careful analysis of the elements and structure of verbal wayfinding instructions. Research in this direction reveals that an ontology for the wayfinding task is needed (Winter, 2002). As an alternative, the concept of wayfinding choremes (Klippel, 2003) can be applied to fit landmarks in the context of each route (Klippel and Winter, 2005).

3.2.2 Graphic design of landmarks

To be effective a navigation system that employs landmarks must enable user to recognize the landmarks used in a route description without significant effort. The presentation of landmarks is subject to a number of constraints implied by the mobile context of use, e.g. the form-factor of mobile devices, the available communication channels and the need to limit cognitive workload of users. To make the use of a landmark based navigation system commercially viable it is also necessary to develop means to derive landmark data efficiently (automatically) from existing information. Route information can be conveyed to the user in various presentation modes that have specifics benefits and shortcomings: E.g. verbal instruction using speech generation leaves the visual channel free, but can be problematic in public environments, textual instructions on the display are private and unintrusive but require high levels of attention while graphical map-like depictions of the situation provide good overview knowledge but can be difficult to interpret. Here we focus on the visualization of landmark information with cartographic instruments to optimize communication in the visual channel.

Obviously, the user's perception and interpretation of visualizations is the key to their effective use. Therefore, the design of visual representations of landmarks should be based on knowledge about user's recognition and interpretation. Designers as well as perceptual psychologists have been studying the recognition and interpretation of visual information by users. Cartographers typically rely on empirical know-how: For conventional 2D maps practical experience over centuries of use has evolved into a collection of visual presentation techniques, design principles and guidelines that are widely accepted by designers (e.g. Bertin, 1973). However, such empirical guidelines are difficult to apply outside their source domain as evidenced by the absence of directly applicable guidelines for the visualization of landmarks and for new forms of geo-visualization (e.g. 3D maps) in general.

Several researchers have examined the impact of different visual designs in navigation applications: Deakin (1996) examined the integration of landmarks into graphic representations or maps for wayfinding purposes and discussed several aspects. The user test conducted with street maps indicates that supplemental landmarks improve navigation performance. In this study two different kinds of landmark portrayals were used: geometric, symbol-like representation and pictorial, stereotype sketches. It was assumed that the stereotype sketches would provoke a strong natural association for the map user and would therefore be more effective than abstract geometric symbols. However, no significant difference between the two presentation styles could be found.A test in the field of car navigation systems by Pauzie et al. (1997) investigated how landmarks could be represented in guidance systems. In their system the background portrayal on the screen was reduced to a turn-by-turn instruction represented by an arrow indicating the next driving action. Two types of pictorial designs were examined: a generic and a specific presentation of the landmark information. The generic pictogram was relevant for all cases belonging to the same category (like church, bridge, park, shop, bank). The specific one represented each landmark object located at the route in a realistic manner. The experiment found that the way the landmarks were presented did not have a strong impact in terms of visual workload. The analysis of a follow-up questionnaire indicated that users preferred a generic portrayal for some of the object categories (church, bar, pharmacy, bridge) while a specific drawing was seen as more useful to represent other objects (bank, fast food, garage, supermarket). The difference depends mostly on the use of trade marks (or logos) as highly familiar elements in the graphics. The study concluded that the recognition and understanding of a landmark is closely linked with its familiarity to the driver (regardless of generic or specific characteristics of its design).

Lee et al. (2001) developed a prototype for visual navigation using a multimedia map. It used photographic images to represent landmarks and matched them directly on a perspective view of the map. Furthermore, full panoramic views from road nodes or sequential photographs along a path were used to provide visual information. The evaluation of the prototype has shown that landmark photographs must be taken from the line of sight in which the object is approached. Therefore, several images are required for each landmark. Additionally, a truly effective landmark photograph should only show the landmark itself and visual clutter like neighbouring buildings have to be removed.

Radoczky (2003) also recommends photorealistic images for the presentation of landmarks, because no generalization operations are needed. The hitch with such an approach is the need for consistency with the real environment, requiring not only appropriate images for different seasons but also updates when structural changes are made to the landmark object.

An alternative approach is to visualize salient objects by means of cartographic generalization. For example important information in a map can be emphasized by using generalization operations like enhancement of the target object itself and simplification or aggregation of the background objects (Sester, 2002).

3.2.3 Aspects of visual cognition

Another source for information on how users interpret what they see is the domain of perceptual psychology, where researchers aim to develop a thorough comprehension of the function of the human visual system. Two prominent theories aim to describe how objects are recognized visually: Image-based object recognition and structure-based object recognition. The first proposes that humans recognize an object by matching the visual image with a snapshot stored in our memory. The second follows the idea that objects are analysed in terms of primitive 3D forms (geon theory) und structural interrelationships (Ware, 2004).

While significant progress has been made in the understanding of individual processing steps within the human visual system, it is currently not possible to derive accurate predictions regarding the effectiveness of visualization techniques from these, as many processes remain active areas of research and complex interdependencies are involved in the whole process that are still little understood.

However, design guidelines can be derived from perceptual psychology research with regards to the (potential) impact of certain visual features like texture patterns, preattentive visual features as well as silhouettes and contours.

Silhouettes as part of the structure-based object recognition play an important role in perceiving the structure of objects. Simplified line drawings are often equal to silhouettes and many objects have particular silhouettes that are easy to recognize. One of the findings of structural theories of perception is that certain simplified views are easier to read. For example, a depiction of a hand can be perceived more rapidly in the form of a simplified line drawing than in the form of a photograph. Studies show that time is needed to perceive details and that simplified line drawings may be most appropriate when rapid responses are required (Ware, 2004).

If the necessary information is not perceivable from the silhouette itself, line drawings are the least effective mode of presentation. Ryan and Schwartz (1956) tested the speed of perception of relevant details in different presentation forms. The four principal illustration modes analysed were photographs of the object, shaded drawings, line drawings and cartoons, which are comparable to cartographic generalized depictions in the sense that the original figure is distorted to emphasize the essential spatial relationships. The time needed to perceive the detailed structure was measured. Recognition of cartoons was shown to be fastest while recognition of line drawings took the longest time. Photograph and shaded drawings were almost equivalent with a perception speed somewhere between the other representations.

The adequate presentation of point information should also take into account ergonomic guidelines for the design of pictorial information (Bruyas et al., 1998): Basic requirements regarding recognition and understanding of symbolic information demand fast and unambiguous understanding of graphical representations. Well-designed pictorial messages enable quick visual information processing in comparison to textual messages. Because of their compactness pictograms are more efficient than textual information in cases where display space is limited. The recognition performance depends on the combination of essential, neutral and additional elements in the pictogram. Essential elements are the typical attributes that are necessary to recognize the object at all, but too much unnecessary detail disturbs the quick understanding of a symbol. Whether confusion of the sign with similar objects occurs depends on the familiarity of the user with the typical attributes of the object. This can be different according to the user's demographic attributes.

For the development of appropriate visual presentation techniques for landmarks and corresponding design guidelines this chapter suggests an approach that builds on existing design and cartographic expertise and insights from perceptual psychology. To identify promising presentation techniques possible options were systematically examined and their suitability evaluated in user studies.

3.3 Types of landmarks

3.3.1 Classification of features types

As part of a master thesis a questionnaire was conducted in which 20 people were asked to describe two different pedestrian routes in the city of Hanover (Lübke 2004). One route leads from the main train station to the main university building, crossing the downtown region with shops and pedestrian areas. The other route leads from a student resident building to the cafeteria of the university, crossing a residential district of the city. Both routes are about 2 kilometres long. The participants were 10 male and 10 female students who have all lived in Hanover for several years. They were instructed to recall the routes from their mind and to write down the wayfinding instructions for a pedestrian unfamiliar with the area. The routes were specified by their start and end points. For both routes the descriptions resulted in a number of different route choices, so not all descriptions have the same content.

The route descriptions were analysed with regards to the landmarks used. All referenced objects were counted and divided into groups of object types. Here five different groups were distinguished: Buildings, monuments (statues), plazas (like market squares or big traffic junctions), references to public transport (underground stations, bus stops, tram tracks) and others (parks, bridges, pedestrian zones, stairs, cemeteries). The distribution of the objects in the route descriptions is shown in Table 3.1.

Table 3.1. Distribution of object types in route descriptions

Object Type	**Route 1 (University District)**	**Route 2 (City Centre)**
Buildings	20 (50 %)	32 (55 %)
Monuments	1 (2,5 %)	6 (10 %)
Plazas	3 (7,5 %)	5 (8 %)
Public Transport	6 (15 %)	7 (12 %)
Other	10 (25 %)	9 (15%)
Total	**40 (100 %)**	**59 (100 %)**

Despite the fact that the routes differ significantly in their environment (Route 2 leads through the shopping area in the pedestrian zone, Route 1 leads through a typical residential area and the university campus), in both routes about 50 % of the referenced objects are buildings. The proportions of the other groups vary slightly. It should be kept in mind that these are only preliminary observations, since only two different routes described by twenty people were examined so no statistically significant statement is possible. Based on the previous research on landmark use and backed by these findings we focus on the visulization of buildings as landmarks. Since most navigation aids are demanded in urban areas, an optimal representation of buildings as landmarks is a central issue.

3.3.2 Characteristics of landmarks

Buildings can be further divided into groups depending on the function or kind of description of the building in the route instructions. For the purpose of this study we distinguish four groups (see Table 3.2). The first group consists of shops and restaurants referenced by their trade name (e.g. H&M, Kaufhof, McDonalds), the second group of other businesses is described more generally with the type of function (like hotel, pharmacy, hairdresser, butcher). A third category is formed by buildings that are referenced by their general function (library, church, university building) or unique name (e.g. Anzeigerhochhaus, Regenwaldhaus). In most cases the proper name is combined with the function (e.g. Luther church), so we combine those. The fourth category covers buildings that are specified by a description of outstanding visual aspects (e.g. the large yellow house, the red clinker brick building).

Table 3.2. Distribution of different building types in route description

Building Type	**Route 1 (University District)**	**Route 2 (City Centre)**
Shop (referenced by name)	4 (20 %)	18 (56 %)
Shop (referenced by type)	3 (15 %)	8 (25 %)
Function / Name	7 (35 %)	6 (19 %)
Visual Aspect	6 (30 %)	0 (0 %)
Total	**20 (100%)**	**32 (100 %)**

If we compare the distribution of objects, it seems that the route environment determines the kind of landmark building description. In the city centre the trade names of shops are preferred, whereas in areas where no trade chains are available other building descriptions using the function or the visual appearance of the object are given. Consequently, it can be hypothesised that the communication and recognition of trademarks is easier than the comprehension of a more complex description of individual visual aspects.

3.4 Designing visualizations

3.4.1 Support for visualization design

Once appropriate landmarks have been selected, the question of how this information can be communicated effectively to the user of a navigation system arises. This is a design issue that involves expertise from a wide range of fields including navigation, visual design, cognitive psychology and mobile device programming. The following steps are necessary to solve such a design issue:

1. The task should be well defined – in the case of landmark visualization this means to define the task "recognize the landmark" in a suitable way. For navigation purposes this means that a user has to match the landmark representation with the corresponding real-world object when he encounters it. This has to be done with sufficient certainty, because the user bases his future locomotion on the result. Landmarks used in another setting, e.g. in a learning scenario where users are expected to memorize a number of landmarks permanently, can lead to different requirements; appropriate criteria for achieving the established goal must therefore be specified.
2. The parameters that influence the design solution should be identified and analyzed. For a landmark-based navigation system this includes various specifications:
   - the delivery medium (e.g. available output modalities could range from an oral description to an animated interactive multimedia presentation on a mobile device),
   - the user (perceptual and cognitive abilities and preferences),
   - the environment and context of use (location, primary task of the user, level of attention, interferences).
3. Potential design solutions should be generated.
4. Potential design solutions should be evaluated and either discarded or refined based on the feedback of the evaluation.
5. A "good" solution that provides a feasible compromise between requirements and practical constraints should be selected for use in the system.

It is obvious that an individual design approach (for each landmark or each user) that starts from the goal of landmark identification and takes the individual characteristics of a specific landmark (or user) into account is not viable for complex systems. Similar problems arise in many areas of visual communication (e.g. visual elements in graphical user interfaces). In order to make the expertise of domain experts, cognitive psychologists and designers accessible in a somewhat systematic way a number of approaches are commonly used:

- A systematic structuring of the design space, according to those variables that can be (realistically) modified, allows designers to create and evaluate design alternatives (see previous paragraph) in a systematic way, reducing the risk of overlooking useful alternatives due to preconceptions or bias. The structuring of the design space encapsulates technical alternatives and has been used successfully to address design problems of significant complexity, e.g. when novel interaction techniques are considered (e.g. Card et al., 1991). Similar approaches, e.g. morphological matrix analysis are used in mechanical engineering and product design.

- Guidelines or style guides are commonly used together with templates, patterns or samples to simplify the creation of potential design solutions (3.) (Tidwell, 2005). Such approaches make a – albeit very restricted – body of design knowledge widely accessible and ensures a certain degree of consistency that is often desired as a global design goal.
- Rules, guidelines or checklists can sometimes be used to provide advice on the selection of design alternatives (4.) (Shneiderman, 2004). Such heuristics are commonly employed in the development of graphical user interfaces and for web pages where they provide a means of making perceptual and design expertise available at an aggregate level that is accessible to designers.

In order to provide useful advice to designers we aim to provide guidelines for the selection of landmark visualizations. Specifically, to produce such design guidelines we have reduced the general design problem as follows:

1. Task definition: We limit our use case to the task of a navigation system in which a user receives route descriptions that employ landmarks at decision points. Thus for our purpose the necessary condition for a landmark visualization is that the user identifies a single landmark at the next decision point with its corresponding real-world object.
2. Parameters and constraints: We assume that the navigation task in which landmarks are employed is a secondary task and thus should be performable by the user with minimal cognitive effort. We also assume that a state-of-the-art mobile device (e.g. Smartphone or PDA) is used as the delivery platform (limiting size of the display and complexity of graphics but supporting raster colour display). As for the user we make no specific assumptions with regards to cognitive or spatial abilities or previous training because we aim at visualizations that should be useful for a general public audience.
3. With regards to potential design solutions we have limited our study to the variation of the level of abstraction of a static visual representation of a landmark as the common denominator that can be implemented on all current devices. This approach also allows seamless integration with other cartographic information as part of the generalization process. The remainder of this section discusses the development of the design alternatives in detail and provides a rationale for the design decisions made in the studied alternatives. Alternative variations, e.g. employing animation, interactive visualizations or other media have not been considered.
4. To provide guidelines for the selection of landmark visualization techniques we have evaluated different design solutions for the depiction of different

landmark categories. The process and the results are described in the following section.

5. We assume that a designer will still make the final selection depending on the specific requirements of the task at hand. We have structured our approach and the resulting design proposal matrix in a way that supports the systematic generation, evaluation and selection of landmarks.This design matrix is conform to the assumptions noted above and can be extended as desired if additional requirements or capabilities need to be addressed.

3.4.2 Developing guidelines for visualization

As the building landmarks fall into four categories, we propose an individually designed visualization style for each group to communicate the landmark information in an appropriate way. This means that the user must be able to recognize the graphics fast and identify its correspondence in the environment easily. Several approaches to the visualization of buildings have been proposed. Some of them are used specially for landmarks, others stem from the field of 3D-City Models: In Lee et al. (2001) cut-outs from photographs are taken and put directly on a map to illustrate the individual facades of landmarks. In contrast to this, non-photorealistic rendering techniques intentionally disregard the idea of images close to reality and present 3D city models in a comic-strip like style rendered by computers (e.g. Döllner et al., 2005). This kind of design is comparable to traditional Bollmann maps and is now often used for tourist maps to present important tourist sights as a 3D-representation on a 2D-map (see Fig. 3.1). A further cartographic technique is to substitute the original object with a map mark whose style may range from mimetic to arbitrary (see Fig. 3.2). If the presentation is shrunk to a point symbol, there are different ways to compose it (see Fig. 3.3): the iconicity of the symbol is very high if it is pictorial designed and very low if it is a geometric, abstract marker (MacEachren, 1995). Pictorial symbols have the advantage to be recognised easily, because no graphics interpretation process is necessary. It is sufficient to match the pattern of the depiction to the environment. This requires that the symbol is not too detailed or confusable (Bruyas et al., 1998). From this point of view logos of trademarks represent pictorial symbols and are therefore useful candidates to depict trade name shop landmarks (see Fig. 3.4).

Fig. 3.1. Tourist map with 3D-tourist sights (taken from tourist map of city Kempten)

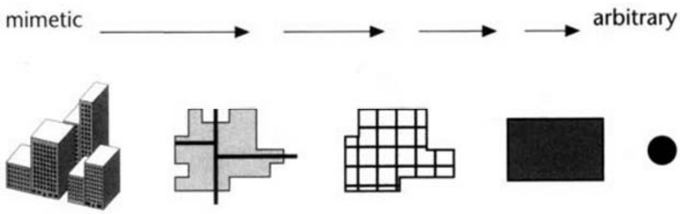


Fig. 3.2. Mimetic to arbitrary continuum of map markers (from MacEachren, 1995, pp.259)

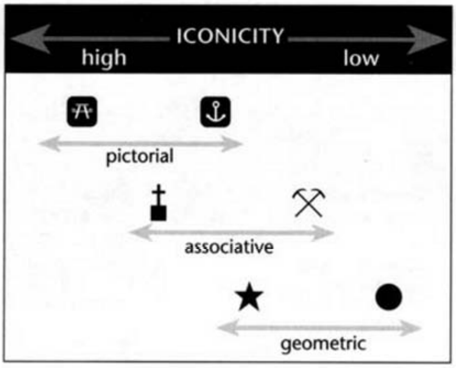


Fig. 3.3. Abstractness of point symbols (from MacEachren, 1995, pp. 262)

Fig. 3.4. Logo icons as landmark representations (cut-out taken from (Klippel 2003))

Altogether, these kinds of depictions form a continuum of different levels of abstractions: on the one hand realistic reproduction (in form of a photographic image or realistic textured 3D-model) on the other hand abstracted presentation as (geometric) symbols or even as words (considering the alphabet as abstract signs) (see Fig. 3.5).

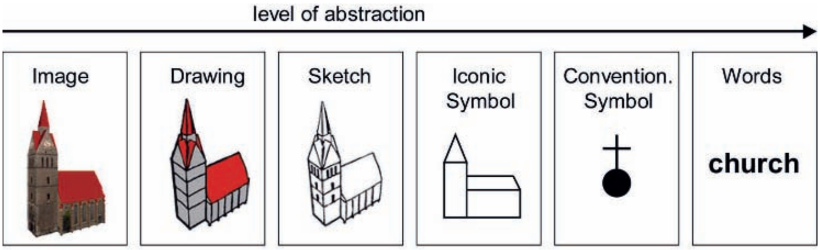


Fig. 3.5. Level of Abstractions for Visualization

A key challenge for map designers is to select appropriate visual presentations while considering secondary design constraints (e.g. desired visual style, restricted color schemes or consistent visual appearance). Also aspects of cartographic generalization have to be taken into account. The image of the original object has to be scaled down to a size suited for a representation in a map. Therefore, some of the conditions under which generalization procedures are conducted in maps also apply here (Shea and McMaster, 1989): congestion (too many features in limited space), coalescence (visible details depend on resolution of output device) and imperceptibility (feature falls below a minimal portrayal size) necessitate the abstraction of the visualization of an object.

To provide designers with a systematic approach we propose to base the visualization of landmarks on different levels of abstractions in order to communicate the different landmark characteristics appropriately. The combinations of landmark types with possible visualization styles span a design space that can be represented as a matrix. In this matrix each landmark type is associated with one or more adequate abstraction levels for their visual representation (see Table 3.3, "+"dicates an adequate match, "(+)" a match that is only valid under specific conditions). The information in this matrix captures experience in practical use and can serve as a guideline to designers. Of course, using words is always possible to convey the information properly, but it may not be the best choice regarding visual and cognitive workload (time needed to process the information) if the verbal information is too complex. Therefore, words are only regarded as appropriate presentation form if there is no better way to convey it with graphical depictions.

A trademark logo is accounted as something generally well known and easy to recognise, so a representative icon is the easiest form to convey the landmark information. Generally, no building description is necessary, but if the building is something (architectural) singular, a sketch with the outline the building may be useful (see Table 3.3, "**(+)**" only valid under this condition**)**. If the shop is only referenced generically, especially designed pictograms or associative symbols are suitable. In cases where there is no appropriate graphics to portray the shop type, words have to be used. Generally, the outline or visual details of the building have little relevance for the landmark information. Specific building functions are often linked to a particular appearance of the building, e.g. typical silhouettes (churches) or size, position and style (town halls and opera buildings are often large, singular buildings, sometimes built in a historic architecture style). Therefore, at least a sketch from the silhouette of the building, sometimes a drawing or image with more details about the façade is needed to recognize the object. The only solution to convey a proper name of a building is to reference it by name with words. If visual aspects are the important facts to describe the landmark, they have to be depicted by a detailed drawing or image of the object.

Table 3.3. Design proposals for landmarks

	Image	**Drawing**	**Sketch**	**Icon**	**Symbol**	**Words**
Shop (Name)			(+)	+		
Shop (Type)				+	+	+
Function/Name	+	+	+			+
Visual Aspect	+	+				

3.4.3 Design examples

The hypothesis of the design matrix needs to be validated by a user test. Therefore, for each building type 3 examples were chosen and visual representations of 3 abstraction levels for each of the 4 building landmark types was generated. Altogether 36 design examples were composed (see Fig. 3.6).

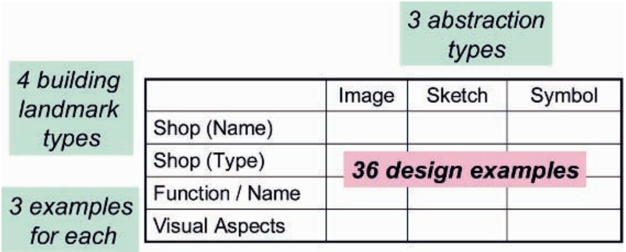


Fig. 3.6. Composing design examples

For creating the abstraction level "image" photographic images of the façades were taken. It is also possible to use 3D-building models of city models if available. The images show only the cutouts of the landmark building. The façade of the landmark buildings were simplified using Adobe Photoshop filters to generate "sketch" level depictions. Hand-drawn sketches were used as an alternative. To create the representations of the third level "icon" the well-known logos of companies were used. General icons like "café" were designed separately. Examples for the 3 abstraction styles are given in Fig.3.7.

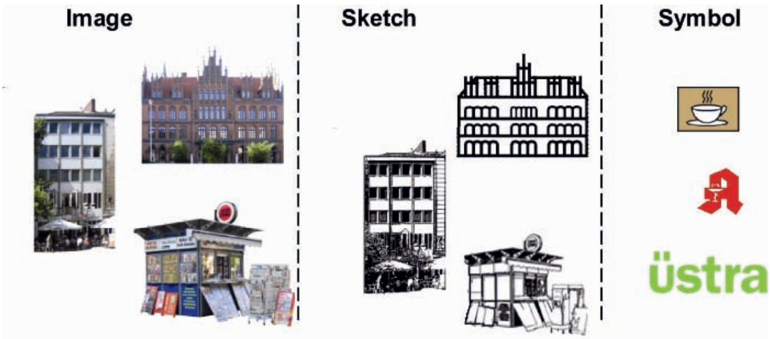


Fig. 3.7. Creating examples in different abstraction levels

As we focus on pedestrian navigation services with mobile maps, we target small PDA and smart phone displays (specifically we used the HP hx4700 as our-reference platform) (see Fig. 3.8). The drafts depict a reduced background map for navigating through a city environment: streets with names and building outlines are given. The colours are reduced to grey scale to improve the figure-ground contrast of the landmark objects. The landmarks are positioned at their original geographic location; therefore parts of the map are overlapped and not visible.

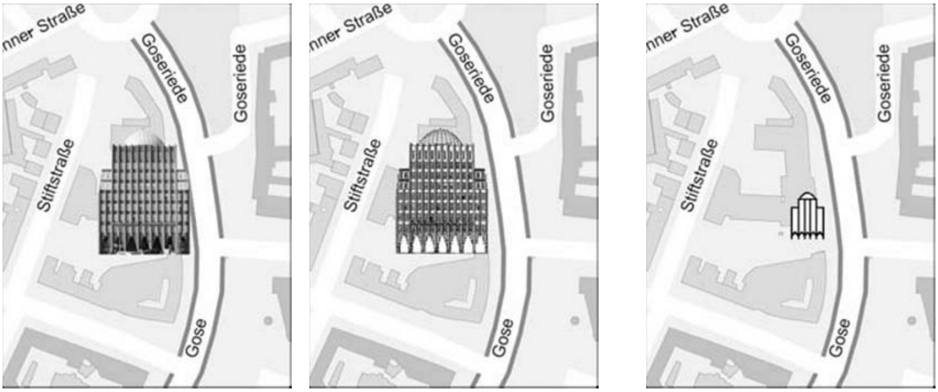


Fig. 3.8. The Anzeiger-Hochhaus building of Hannover: Image (left), sketch (centre) and symbol (right)

These visualization drafts were then presented to test users to check if the kind of depiction is recognizable and able to convey the landmark information completely. The results allow the comparison of the relative usefulness of different landmark presentations and can serve as the basis to improve the design matrix for the visualization of building landmarks.

3.5 Evaluations

3.5.1 Approaches to evaluation and user test

To examine how the different landmark visualization designs perform for different landmark categories the different design options have to be evaluated with real users. The results of these findings are then provided as a design matrix to support designers in selecting appropriate representations for the task at hand.

A wide range of evaluation techniques has been developed in the domain of human-computer interaction and is available to designers of visualization techniques (e.g. Nielsen, 1994; Preece et al., 2002). For our purposes of evaluating visualization techniques for landmarks only tests in which end users are involved are suitable to provide the necessary data. The most common methods for conducting such user test are think-aloud protocols (or cooperative evaluation), wizard-of-oz tests, usability tests (both in laboratory settings and in the field) and questionnaires.

In think-aloud protocols users are observed performing a number of (scripted) tasks with a prototype system that uses the visualization technique under consideration and are asked to explain what they are doing and why. An evaluator records the interaction and notes problems. In cooperative evaluation the evaluator is actively involved in the evaluation and asks “why?” and “what if?” questions. The evaluator also provides clarification and advice if problems arise or parts of the functionality are not implemented in the evaluated system. An advantage of the think-aloud techniques is that they can be used with partially functioning prototypes as well as implemented applications. They provide quick feedback, are relatively inexpensive to conduct and suitable for exploratory design approaches. They can provide evaluators with insights into the way users think and work with a visualization that can be used to refine user requirements. A potential problem is that the user interaction experience can be influenced by the presence of the evaluator and an inherent shortcoming is that the collected information is subjective; no quantifiable measures are obtained, making it difficult to compare results between several alternatives. In practice test user often have difficulties explaining their thoughts and actions while interacting. Thus, while think-aloud protocols can be a useful tool for a designer to validate his landmark designs it is not suitable to derive general design information as is desired for landmarks.

In wizard-of-oz tests users perform test tasks by interacting with a visual mock-up of the visualization. An experimenter, the „wizard“, simulates the missing functionality of the system. The interaction is recorded and analyzed. Using this approach, user interactions can be observed and recorded in detail. The resulting tests are often more realistic than think-aloud protocols and it is possible to obtain certain objective measures with regards to the usability of the visualization (e.g. task times, number of actions performed, number of errors). However, wizard-of-oz tests are not without problems, as they require significant efforts and expenses to prepare and conduct as well as an experienced “wizard”. Given the impact of the wizards interaction on the results it can be difficult to establish controlled test conditions and thus difficult to compare results between tests.

In user tests users perform test tasks by interacting with a working implementation or prototype of the visualization. The interaction can either take place in a usability laboratory or within the environment in which the system will be used in practice. This interaction is recorded and analyzed. Benefits of this approach are the high degree of realism that can be achieved, the possibility to capture problems that arise through complex interdependencies in real-world settings and the provision of objective usability measures (e.g. task times, number of actions performed, number of errors). In laboratory settings controlled test conditions can be established which makes it possible to compare results across and between tests. Drawbacks involved in the practical implementation of user test are the fact that they require significant efforts and expenses to prepare and conduct and that they are seldom suitable to test early design concept prototypes. Care is required to establish a realistic but controlled environment and the equipment required for recording objective measures and video of the interaction can be another limitation, especially if tests in a mobile setting are desired.

Questionnaires are popular to study how users experience different designs and can provide insights into preferred features, preferences and other subjective usability ratings like satisfaction. They are often used in combination with other evaluation approaches or after test sessions. After their interaction with the visualization users fill in a questionnaire that queries their subjective opinions about several usability metrics. A benefit of questionnaires is that they are fairly quick and inexpensive to create, relatively simple to conduct and can provide feedback based on real user experience. Key problems with questionnaires are that they typically do not identify specific problems of a design and that response rates are often poor.

All these techniques can be employed by designers of navigation systems and should in fact be used to validate a design based on the visualization techniques proposed here. While some standard usability measures (e.g. accuracy, error rate, cognitive workload, subjective satisfaction) apply directly to the landmark visualizations, others (e.g. learnability, efficiency) can only be reasonably measured within a complete application so that high-level validation of the integrated design is mandatory. More detailed descriptions of the individual techniques can be found in the HCI literature (e.g. Dix et al., 2003; Mayhew, 1999).

For our test we have decided on an approach that combines elements from wizard-of-oz tests with user tests and questionnaires as described in the following sections to obtain results with real-world users under controlled conditions for the simple landmark recognition task.

3.5.2 User test of the design examples

To evaluate the quality of the design examples a user test was planned and carried out (Kuhnt, 2005). To evaluate different aspects of design examples, the test was structured into three different parts (see Fig.3.9). In the first section called "interpretation" all 36 design examples were shown to the participants on a PDA device. The subjects were asked to describe what they see, the answers were recorded. Additionally, it was rated how fast the answer was given. Instant responses were categorized as fast, answers given after a few seconds of reflection were categorized as slow.

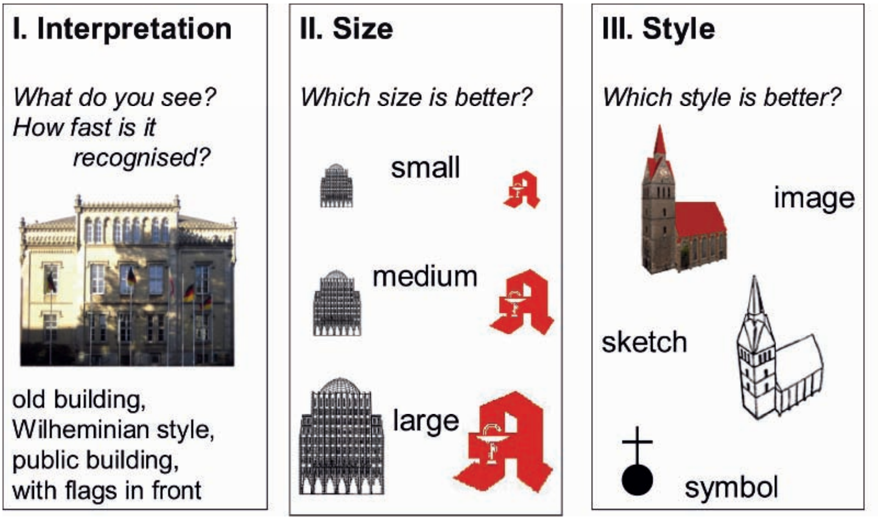


Fig. 3.9. User test outline

In the second part of the questionnaire the participants were asked to assess the optimal size of the landmark presentation. This is an important issue, because the PDA display size is limited and therefore the background map (which provides global position and orientation) competes with the details of the landmark visualization. Especially when photographs are given as landmark portrayals, the image has to be presented in a specific size to show the details of the object. But the larger the landmark object is portrayed, the more of the background map is masked. Therefore, it is necessary to determine if there is an optimal size for the landmark visualization depending on the landmark type to convey both – background and landmark - appropriately.

For this part of the test 3 different sized presentations of landmark objects were put on the same background map and then presented (on paper prints) at the same time (see Fig. 3.10). The subjects were asked which presentation size they prefer. Design examples for each building landmark type were chosen for this task to check if the preferred size of the landmark visualization depends on the portrayed landmark type.

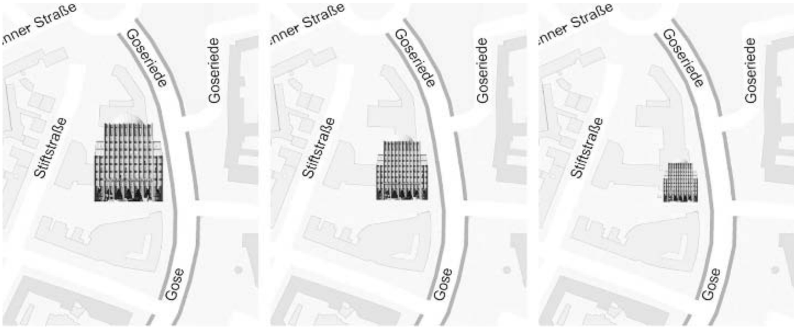


Fig. 3.10. Proposed sizes for landmark visualization (on PDA with 640 x 480 pixel display): Large with ca. 130 x 170 pixel (left), medium with ca. 100 x 130 pixel (center) and small with ca. 70 x 90 pixel (right)

In the last section of the user test the three different abstraction styles designed for one landmark object were compared. The participants were asked to choose the presentation style they thought best suited for the given building type. The test objects were taken from each building landmark category because we assumed that the appropriate style depends on the group the building landmark belongs to.

3.5.3 Results of user test

The questionnaire was separately conducted with each subject. It took about 25 minutes each, including an introduction to the task. Altogether 20 subjects took part in the interview: 9 males and 11 females in the age of 18 – 57 years (on average 31 years old). 15 participants were familiar with the city of Hannover, 5 of them were not. All were of German nationality.

The results of the interpretation part show that there are different verbal descriptions for the same landmark visualization. Because the complex analysis is irrelevant for the work presented here, the results of that part are excluded here. Only the outcome of the speed of interpretation categorization is presented. In Fig. 3.11 the results are shown in diagrams for each building landmark type. Next to each diagram a prototype for the group is referenced (like “McDonalds”) to ease the understanding of the different groups. In the diagrams it is shown which kind of presentation style (image, sketch or symbol) leads to a fast or slow response (referred to as “speed of perception”).

Looking at the “shop (name)”-group it is clearly visible that the presentation style symbol leads to a fast interpretation by most of the subjects. The same holds for the “shop (type)”-group. On the other hand, “function/name”-group visualizations are rapidly interpreted in all three styles, whereas “visual aspect”-group objects need time for the interpretation process in all cases.

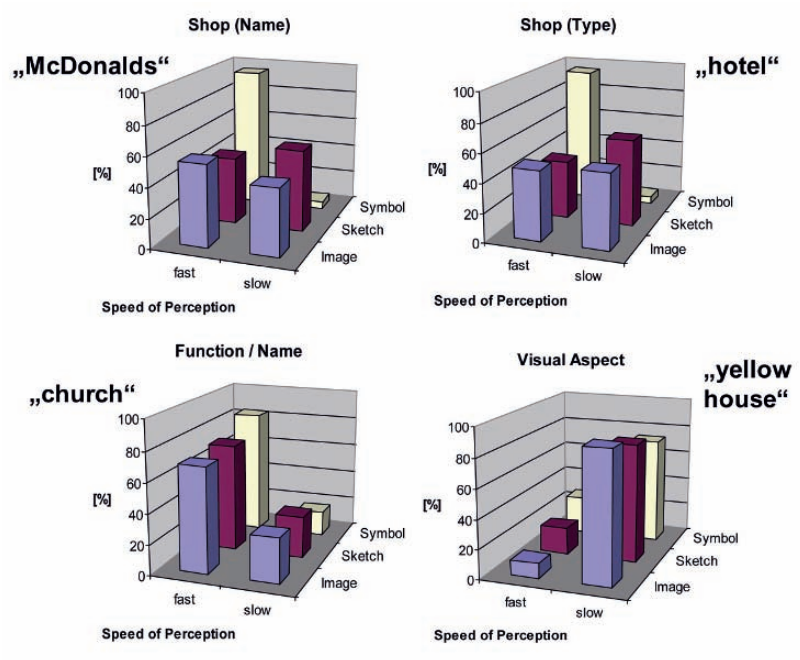


Fig. 3.11. User test part I “interpretation”: speed of perception

The results of the second part show that the preferred size of the visualization is clearly differing between the symbol style on the one hand and the sketch and image style on the other hand (Fig. 3.12). For a symbol presentation only a small size is required, as nearly all participants selected this size. For the sketch and image style the medium size is rated best, but also the small size was chosen by a large group of subjects. It is interesting to observe that the results of the groups sketch and image style are comparable to each other. So, regarding this issue no difference has to be made between these two presentation forms.

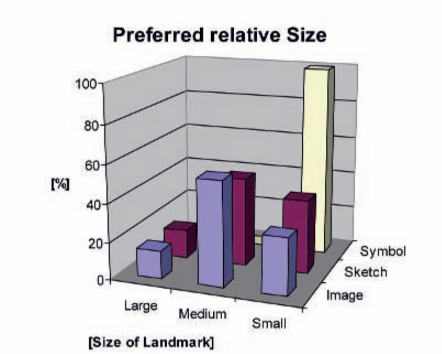


Fig. 3.12. User test part II: User preferences for scale of the landmark portrayal.

In the third section of the test the participants were requested to valuate the presentation style for landmark objects (Fig. 3.13). The ratings “good”, “acceptable” and “poor” for the valuation of style are given. For landmarks belonging to the group “shop (name)”, a symbol style is rated as a very good presentation form, while the portrayal in form of images or sketches is rated as poor. The same finding holds for the “shop (type)”-group. But here the rating “acceptable” for the image or sketch style is chosen more often. On the other hand the rating for the “function/name”-group clearly show that the image portrayal of the landmark is valuated as best. But also the sketch and symbol style seem to be appropriate for many participants. Completely different are the results for the “visual aspect”-group. Unambiguously, the symbol presentation form is not acceptable for the most of the subjects. Most of them rank the sketch form as the most suitable presentation style for this group.

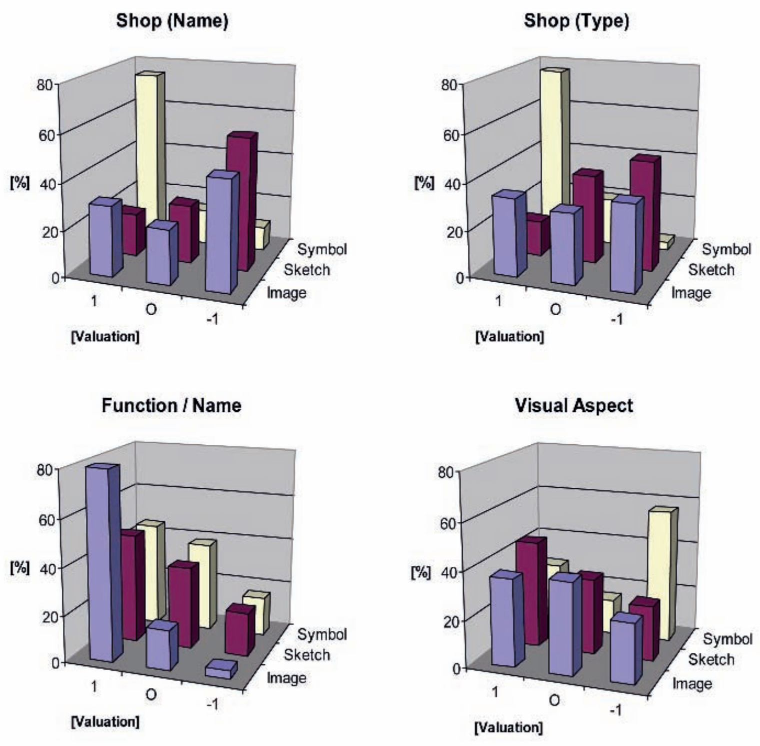


Fig. 3.13. User test part III: Valuation of the presentation style. 1: good, 0: acceptable, -1: bad.

The user test has been carried out using only a small number of design examples for each building landmark group. Because of that, the results have to be handled with care and more design examples have to be tested to consolidate the findings.

So far the test results support the proposed design matrix (see Table 3.3) and some interesting trends can be observed:

- well-known logos (like the McDonald icon) are easy to interpret, accepted by the users and therefore a very good landmark presentation form
- the optimal presentation of visual aspects is still missing
- the optimal presentation size differs between logos and more detailed presentation forms (like sketches or images) which need more place
- for buildings with a specific function or name a more detailed presentation form seems appropriate

3.6 Conclusion and outlook

The approach presented here proposes a design matrix as a visualization guideline for landmarks. We have examined the different feature types that are useful as landmarks. We have found that about 50 % of all landmarks used in common wayfinding instructions are buildings and identified 4 different categories of building landmarks: well-known shops (trade chains), shops referenced by their type, buildings with specific names or functions and buildings described by their visual appearance. For the visualization of landmarks from each of these categories the impact of different abstraction levels in visual design were examined and evaluated by a user test. The results lead to design recommendations that are captured in a design matrix that proposes different levels of abstractions as appropriate visualizations for different categories of building landmarks.

As a further outlook this knowledge could eventually be used in an automatic tool to provide designers with advice or provide a set of rules to produce the visualizations automatically from databases. Further work is necessary to understand the dependencies between users und preferred visualization. With this it would be possible to automatically adapt the visual presentation to a user and his specific task at runtime.

Finally, the building landmarks discussed here represent only half of all landmarks used in common wayfinding descriptions. The extension of the approach to other types of landmarks is therefore another obvious task for future work.

Acknowledgements

The execution and analysis of the user test was part of a diploma thesis done by Sascha Kuhnt at the institute of cartography and geoinformatics, University of Hannover. We gratefully acknowledge the comments of the reviewers.

References

Bertin, J. (1973): Graphische Semiologie, de Gruyter Verlag.

Bruyas, M-P., Le Breton, B. and A. Pauzie (1998): Ergonomic Guidelines for the Design of Pictorial Information, *International Journal of Industrial Ergonomics* 21, pp.407-413.

Card, S. K., Mackinlay, J. D., and Robertson, G. G. (1991): A morphological analysis of the design space of input devices. *ACM Trans. Inf. Syst*. 9, 2 (Apr. 1991), pp. 99-122.

Daniel M.-P. and M. Denis (1998): Spatial Descriptions as Navigational Aids: A Cognitive Analysis of Route Directions, *Kognitionswissenschaft* 7(1), pp. 45-52.

Deakin, A. (1996): Landmarks as Navigational Aids on Street Maps, *Cartography and Geographic Information Systems* 23 (1), pp. 21-36.

Denis, M., Pazzaglia, F. Cornoldi, C. and L. Bertolo (1999): Spatial Discourse and Navigation: An Analysis of Route Directions in the City of Venice, *Applied Cognitive Psychology* 13, pp. 145-174.

Dix, A, Finlay, J., Abowd, G., Beale, R. (2003): Human Computer Interaction, Prentice Hall.

Elias, B. and M. Sester (2002): Landmarks für Routenbeschreibungen, in: GI-Technologien für Verkehr und Logistik, Beiträge zu den *Münsteraner GI-Tagen* 20./21. Juni 2002, IfGI prints, Band 13, Institut für Geoinformatik, Münster.

Elias, B. (2003): Extracting Landmarks with Data Mining Methods, in: Kuhn, W., Worboys, M.F. and Timpf, S., (Eds.): "Spatial Information Theory: Foundations of Geographic Information Science", Vol. 2825, *Lecture Notes in Computer Science*, Berlin, Springer, pp. 398-412.

Elias, B. and C. Brenner (2004): Automatic Generation and Application of Landmarks in Navigation Data Sets, in: Peter Fisher (Edt.): Developments in Spatial Data Handling - *11th International Symposium on Spatial Data Handling*, Springer, pp. 469-480.

Elias, B. (2006): Extraktion von Landmarken für die Navigation, Dissertation, *Deutsche Geodätische Kommission, Reihe C*, Number 596, München.

Döllner, J., Buchholz, H., Nienhaus, M. and F. Kirsch (2005): Illustrative Visualization of 3D City Models, *Proceedings of SPIE - Visualization and Data Analysis* (VDA 2005), San Jose, CA, USA.

Gartner, G. and S. Uhlirz (2001): Cartographic Concepts for Realizing a Location Based UMTS Service: Vienna City Guide LoL@, *Proceedings 20th International Cartographic Conference,* 6.-10. August, Beijing, China.

Golledge, R. (1999): Human Wayfinding and Cognitive Maps, in: Wayfinding Behavior, John Hopkins Press, pp. 5-46.

Horn, M and Köpke, J. (2005): Navigation mit Landmarken (für PDA). Seminar paper (unpublished). Institute of Cartography and Geoinformatics, University of Hannover.

Klippel, A. (2003): Wayfinding choremes. Conceptualizing wayfinding and route direction elements. Phd Thesis. University of Bremen.

Klippel, A. and S. Winter (2005): Structural Salience of Landmarks for Route Directions, In: Cohn, A.G.; Mark, D.M. (Eds.), COSIT 2005. *Lecture Notes in Computer Science*, 3693. Springer, Berlin, pp. 347-362.

Kray, C., Laakso, K., Elting, C. and V. Coors (2003): Presenting Route Instructions on Mobile Devices, *Proceedings of the Intelligent User Interface*, ACM, pp.117-124.

Kuhnt, S. (2005): Kartographische Visualisierung von Landmarken, Master thesis (unpublished)., Institute of Cartography and Geoinformatics, University of Hannover, 2005.

Lee, Y.C., Kwong, A., Pun, L. and A. Mack (2001): Multi-Media Map for Visual Navigation, *Journal of Geospatial Engineering* 3(2), pp.87-96.

Lovelace, K., Hegarty, M., and D. Montello (1999): Elements of Good Route Directions in Familiar and Unfamiliar Environments, in: C. Freksa and D. Mark (Edt.): *Spatial Information Theory: Cognitive and Computational Foundations of Geographic Information Science*, Springer, pp.65-82.

Lübke, C. (2004): Extraktion von Landmarken aus ATKIS-Daten. Master thesis (unpublished). Institute of Cartography and Geoinformatics, University of Hannover.

MacEachren, A. (1995): How Maps Work, The Guildford Press.

Mayhew, D. (1999): The Usability Engineering Lifecycle, Morgan Kaufmann Publishers.

Nielsen, J. (1994): Usability Engineering, Academic Press.

Pauzie, A., Daimon, T., Bruyas, M.-P. and R. Trauchessec (1997): How to design landmarks for guidance systems?, *Proceeding (CD-ROM) of the 4th World Congress on Intelligent Transport Systems* (ITS), Berlin.

Preece, J., Rogers, Y., Sharp, H. (2002): Interaction Design. *Beyond Human-Computer Interaction*, John Wiley and Sons.

Radoczky, V. (2003): Kartographische Unterstützungsmöglichkeiten zur Routenbeschreibung von Fußgängernavigationsystemen im In- und Outdoorbereich, Master thesis. Vienna University of Technology.

Radoczky, V. and G. Gartner (2005): Extent and Effectiveness of Map Abstraction for Communicating Routes in a LBS, in: *Proceedings of 22st International Cartographic Conference*, 9. - 16. July 2005, La Coruña/Spain.

Raubal, M. and S. Winter (2002): Enriching wayfinding instructions with local landmarks, in: M. Egenhofer and D. Mark (Eds.): *Geographic Information Science*. Vol. 2478 of Lecture Notes in Computer Science, Springer, pp.243-259.

Ryan, T. and C. Schwartz (1956): Speed of Perception as a Function of Mode of Representation, *The American Journal of Psychology* 69, pp.60-69.

Sester, M. (2002): Application Dependent Generalization - The Case of Pedestrian Navigation, *IAPRS* Vol. 34, Part 4 "Geospatial Theory, Processing and Applications", Ottawa, Canada.

Shea, K. and R. McMaster (1989): Cartographic Generalization in a Digital Environment: When and How to Generalize, *Proceedings Auto-Carto 9*, Baltimore, Maryland, pp.56-67.

Shneiderman, B. (2004): Designing the User Interface, Addison Wesley.

Tidwell, J. (2005): Designing Interfaces: Patterns for Effective Interaction Design, O'Reilly.

Tversky, B. and P. Lee (1999): Pictorial and Verbal Tools for Conveying Routes, in: C. Freksa and D. Mark (Edt.): *Spatial Information Theory: Cognitive and Computational Foundations of Geographic Information Science*, Springer, pp. 51-64.

Ware, C. (2004): Information, Visualization – Perception for Design. 2nd Edition, Morgan Kaufmann.

Winter, S. (2002): Ontologisches Modellieren von Routen für mobile Navigationsdienste. In: Kelnhofer, F.; Lechthaler, M. (Eds.), TeleKartographie und Location-Based Services. *Schriftenreihe der Studienrichtung Vermessungswesen und Geoinformation. Technische Universität Wien*, Vienna, pp. 111-124.

4 An Incremental Strategy for Fast Transmission of Multi-Resolution Data in a Mobile System

Jean-Michel FOLLIN[1], Alain BOUJU[2]

[1] Laboratoire L2G, ESGT – CNAM, Le Mans
[2] Laboratoire L3i, Université de La Rochelle

Abstract. In this chapter, a model for management of vector multiresolution geodata in a client-server framework is proposed. Its specific focus is to take into account the constraints related to mobile context (limitations of storage capacity and transfer rate). In particular, the amount of data exchanged between client and server can be minimized by reusing the data already available on the client side with the concept of "increment". An increment corresponds to a sequence of operations allowing the reconstruction of an object in one resolution from another consecutive resolution of the same object available in the client's cache. Increments are computed from a Multi-Resolution database and stored on the server side. Interest in using increments depends on features of a data set's different resolutions like the proportion of shared objects. This strategy is validated with theoretical cost and two simulations (with and without) transfer. It allows speeding up the access to multi-resolution data for a mobile user.

4.1 Introduction

In this chapter, we propose a solution to manage multi-resolution geographical data in a mobile system. After an overall presentation of principles already seen in our preceding papers (Follin et al., 2005a; Follin et al., 2005b), we provide some improvements and new evaluations of our strategy.

First we give an overview of the existing solutions. We cite works like the GiMoDiG Project before presenting basic conditions and choices on which is founded our incremental solution based on a Levels of Detail (LoD) approach.

Then we give presentation of our concepts of LoD object, increment and of our data management principles based on the reuse of data from a local cache. Both polylines and regions can be reused from one LoD to another consecutive one with the increments.

In the third part we present aspects related to our incremental strategy: creation of generalisation and refinement increments from two consecutive existing levels on the server-side, and their use in order to reconstruct object geometries on the client-side. In particular the increments are only computed for objects which respect an "efficiency" condition.

Afterwards an implementation and a validation of our approaches are presented. A multi-resolution dataset is produced by using operators adapted to incremental strategies (i.e. with “good” proportions of shared objects and geometries) and preserving data topological consistency. The validation is made through two types of simulation (with real data from maps and GPS): a “global” and a “scenario-oriented” one. In the first case, we consider the complete transformation of a layer at a certain LoD in another representation of the same layer at a consecutive LoD. In the second case, evaluation is made through the simulation of transfer between a mobile user making zoom and pan operations and a data server. Finally we conclude on our incremental strategy and present outlook.

4.2 Some solutions for managing multi-resolution data in a mobile context

Large amount of geodata can be delivered across the Internet with the establishment of online data clearinghouses and faster network protocols (Buttenfield, 2002). Furthemore many available datasets have reached a very high resolution and their size has dramatically increased (Bertolotto et al., 2001). As they can be over-detailed with respect to the user’s need, different works have focused on the methods to transmit the resolution of data the more adapted to the user’s query in a mobile or web context. Two global strategies are possible to provide multi-resolution geodata through a limited communication channel (Cecconi et al., 2002): Real-Time Generalisation (RTG) and Levels of Detail approach.

4.2.1 Real-time generalisation and LoD approach

The real-time generalisation can also be called on-demand, on-the-fly, on-line generalisation or more generally *process-oriented approach*. It aims to produce in real-time a temporary and generalised dataset from a more detailed one (Fig. 4.1.A). In this way it allows avoiding redundancy of data (van Oosterom, 1995).

Representations of data correspond to different resolutions in our works and a resolution of data is appropriate to a particular scale of representation. Therefore representation, resolution and scale are considered as synonymous to each other in our context.

In the LoD approach, different resolutions of data are pre-computed on the server side. They are stored in a Multi-Representation Data Base (MRDB) and only the most adapted level of detail is sent to the client (Fig. 4.1.B). MRDB can be produced by acquiring information at various scales from different sources or by deriving different LoDs from one dataset with generalisation operations. The first solution (*representation-oriented approach*) poses maintenance problems because updating of a specific level introduces inconsistencies between different representations. Such issues can be solved with the second one (*derivation-oriented* *solution*) where links can be established between the different representations of the same real-world objects: these relations allow the propagation of all updates from base data to the other LoDs (Hampe et al., 2003):

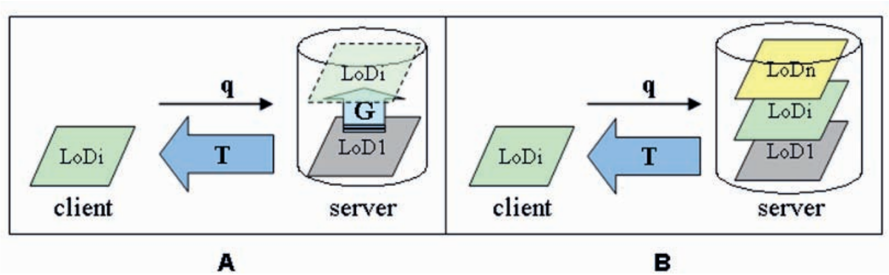


Fig. 4.1. Main solutions for the management of multi-resolution data in a mobile or web context

These methods have advantages and drawbacks (Cecconi, 2003). In a LoD (or multi-representation) approach, less calculation power is required on the server side and the access between data levels is faster for the client than in a RTG one. Furthermore better quality maps can be produced: more complex and computationally expensive algorithms can be used for the generalisation when this one is accomplished off-line. But there is a need of large storage capacity on the server side. On the other hand, data can be provided to the user with more flexibility with RTG because there are no fixed levels. But a balance must be found between the response time and the map quality.

4.2.2 Concrete examples

In the research area on RTG, works have focused on hierarchical data structures. Thus van Oosterom has proposed specifical structures, called reactive data structures, in order to speed up the generalisation process while reducing the memory use (van Oosterom, 1995): the BLG-Tree, the Reactive Tree and the GAP Tree.

A RTG approach can be combined with a multi-resolution database in order to provide data on a larger range of scales (Lehto et al., 2001). In the GiMoDig project some precomputed LoD are used in order to minimize the effort of computation work during the RTG process (Sester et al., 2004a). RTG produces faster the resolution requested without generalising everything by selecting the most suitable MRDB level to its input data. Morphing operators can be used in such mixed approaches (van Kreveld, 2001; Cecconi, 2003) because they compute intermediate states from a starting and a final one, i.e. between two LoDs (Fig. 4.2). A model for on line generalisation which combines access to multi-scale data with dynamical processes for conflicts resolution and visual rendering is proposed in Jones et al. (2000).

The limitations of simple use of RTG are pointed out by these approaches. Emerging fields of progressive transmission of vector data aim to overcome them.

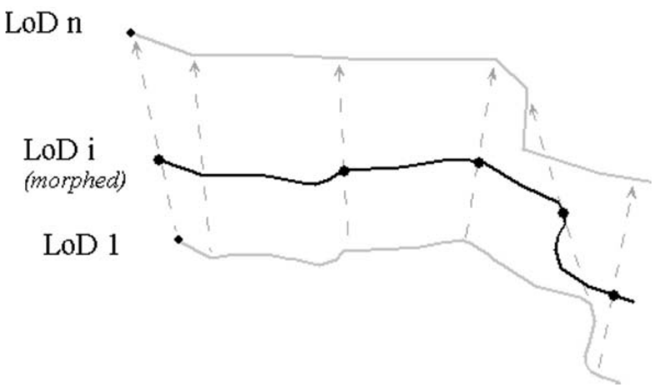


Fig. 4.2. Creation of an intermediate state LoD i from two representations at LoD n and LoD 1 (based on Cecconi, 2003)

The first works on progressive transmission of vector data are those of Bertolotto (1998), Buttenfield (1999) and Oh et al. (1999). They have been inspired by researches in different domains:

- transmission of raster data with interlaced GIF (*Graphic Interchange Format*) and progressive JPEG (*Joint Photographic Expert Group*) formats, and
- transmission of vector TINs[1] data in computer graphics.

Hoppe (1996) describes a scheme for the simplification of triangulated surfaces with progressive meshes: meshes are incrementally transmitted through an internet connection. This procedure is usefull for the highly detailed geometric models but is not adapted to all types of vector data. In Bertolotto et al. (2001) and Zhou et al. (2005) a model for structuring vector data at different resolutions is proposed and implemented. It is based on a hierarchical tree structure.

Similar approaches have been proposed by Buttenfield (2002) and Yang (2005). This model is illustrated in Fig. 4.3. A compressed format adapted to the streaming of multi-resolution geodata, RaveGeo©, is presented in Persson (2004). This work has pointed out the necessity to maximize number of shared points between different geometries in order to adopt an incremental approach.

A continuous generalisation method is proposed in Sester et al. (2004b). It is based on incremental transfer of operators, the Elementary Generalisation Operators (EGO's), and aims to avoid "popping effects" encountered during a continuous zoom. A formalism is proposed for representing progressive transformation of the more detailed representation P^n of a polyline into its coarser one P^m:

[1] Triangular Irregular Networks

$$P \equiv P^n \equiv P^{i_0} \xrightarrow[g_0]{} P^{i_1} \xrightarrow[g_1]{} \ldots \xrightarrow[g_{k-1}]{} P^{i_k} \equiv P^m$$

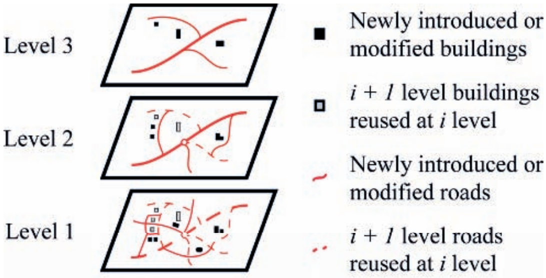


Fig. 4.3. Efficient encoding of a map sequence with multiple representations at three levels of detail (based on Bertolotto et al., 2001)

4.2.3 Incremental strategy for fast transmission

Basic conditions and choices

A mobile system is developed in the "Informatics Image Interaction" Research Laboratory (L3i) in the form of a Java applet. It is mainly conceived for navigational purpose and is limited to presentation of road network and buildings. We use the spaghetti vector model for data storage and management because it is a good choice for a visualisation system with low capabilities.

Our solution is based on the following constraints:

- Geodata are served from one single and topologically consistent source, the server, and several clients can be processed at the same time by one server.
- Data transfer from server to client is only performed as an answer to a client request.
- We just consider the requests which take into account only a geometrical condition: intersection of Multi-Resolution (MR) dataset with a visualization window (conditions on attributes are not considered).
- Data management is possible on the client side for data already available in cache (display in the case of connection break) and those providing from GPS (processes related to visualization and location)
- Geographical data have to be aesthetically pleasant (i.e. map must be readable and clear) and topologically consistent (i.e. relations between objects must be respected at all resolutions).

By considering these conditions, we have chosen a multi-representation approach, more especially a derivation-oriented one. From the cartographic point of view, this solution provides maps with good aesthetic and topological qualities. From the management point of view, we aim to reduce the amount of transferred MR data between the client and the server. For that, the server side process is performed in two steps and accomplished each time base data are updated:

- first, some generalisation operations are executed for deriving coarser cartographic representations from a base dataset,

- secondly, increments are computed between the different consecutive levels of detail and stored on the server.

For this second step we have to consider the changes to perform on one object's representation in order to rebuild a consecutive resolution of the same object in the generalisation and in the refinement direction.

Adopted generalisation processes

The chosen generalisation operators are those which allow to:

- preserve topological consistency and keep the global aspect of objects in order to produce a "good" generalisation,
- keep a part of object geometry in order to be adapted to an incremental strategy.

Roads and buildings are considered as essential features in a mobile system for navigational purpose. Our incremental strategy has been applied to two particular cases: polylines representing roads simplified with a modified version of the well-known Douglas-Peucker (DP) algorithm and regions representing buildings generalised with the method of "minimal length of facade" (Sester et al., 2004b).
From the implementation point of view, our strategy has been validated on road data covering the city of La Rochelle and on several examples of buildings.

Simplification of a polyline with the classic Douglas-Peucker algorithm (and with all similar "vertex sub-sampling" algorithms) results in a selection of a subset of polyline's vertices (Saalfeld, 1999). It retains only "critical points" that define the essential shape of the line (Fig. 4.4). Simplified representation is more or less close to the original according to a chosen tolerance value. We have modified the DP algorithm in order to respect topological relations between connected roads.

Furthermore a contraction operator has been applied to the traffic circles in order to replace them by points. So on some polylines extremities have been moved in order to respect the network's connectivity (cf. section 4.5.1).

Selection/elimination process has been applied to roads set in order to remove objects according to their importance.

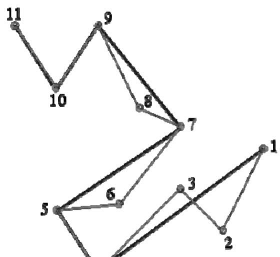


Fig. 4.4. Application of Douglas-Peucker algorithm on a polyline

Building simplification is more a structure reduction algorithm than a strictly point reduction method (Sester et al., 2004b). Indeed properties like parallelism and rectangularity have to be respected in this algorithm that eliminates too short façade elements. Three different kinds of structures can be identified, for which appropriate reduction methods are defined: extrusion or intrusion, offset, and corner (see Fig. 4.5).

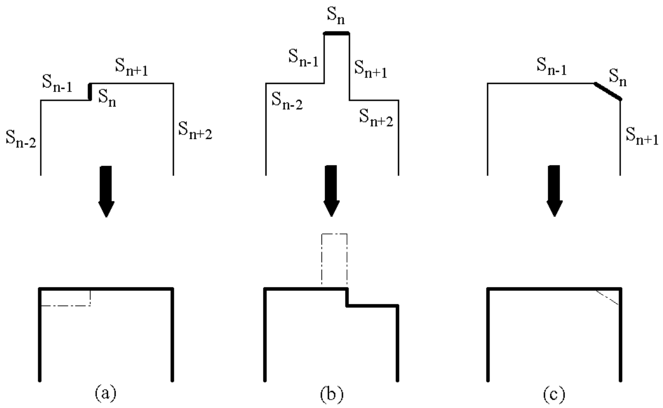


Fig. 4.5. Elimination of short facade S_n: offset (a), intrusion/extrusion (b) and corner (c) (based on Sester et al., 2004b)

These operators are interesting in a client-server context with limited resources by reducing the amount of transferred data. Generalised dataset can be sent before a more detailed one. Furthermore such operators guarantee the "sharing of geometries" which is essential in an incremental strategy.

Consequences on incremental transitions between different levels of detail

If we observe results of these operators on different representations of regions and polylines, we can deduce the different changes to perform on an object in order to rebuild its more generalised or detailed representation.

Case of simplified polylines: In consequence of the simplification process, eliminated points need to be inserted in the generalised representation of the polyline in a refinement transition and must be removed from the most detailed one during a generalisation transformation. In this last case a choice must be done between conservation of shared points or removal of details. Moreover, vertices can be moved between different LoDs, for example in order to respect the topological relations with neighboring objects: coordinates of these points must be changed during a refinement or generalisation transition.

Case of simplified regions: In generalisation transition, vertices can be either kept (for the common points), removed (for the details), or moved (for preserving parallelism and rectangularity properties of building). In refinement transition, vertices can be either moved or introduced (for adding details).

These generalisation operators are expected to be performed on the server side and followed by a process of increment creation. A formalism has been defined in order to consider different object resolutions and transformations between them.

4.3 MR data and MR data transfer models

4.3.1 Data model

A multiresolution data model adapted to limitations of mobile context has been defined in Follin, et al. (2005b). The data organization is based on the traditional definition of a geographic map: objects are grouped into layers and a sequence (or overlay) of layers forms a map (Tomlinson, 1967). As representations of objects vary according to the level of detail, we consider different LoD objects grouped into different LoD layers. Increments allow navigation through these different LoD objects and in this way reuse of available LoD representations on the client-side. In order to reduce volume of data transferred from server to client, increments are sent if their size is less important than the size of LoD objects.

Layer and object
A layer, noted *l*, is a collection of objects associated with a description of their attributes. Each layer corresponds to a specific theme (e.g. transportation network or buildings) that can be decomposed in different LoD layers.
A map is defined as a succession of thematic layers which aims to be manipulated and visualized at a specific scale (Follin, et al. 2005b).
An object entity is defined by the quadruplet (o, t, g, γ) where:

- *o*: unique identifier,
- *t*: last time of modification (timestamp value),
- *g*: location and geometrical description (modelled by one among six two-dimensional geographical objects of spatial domain G : Point, Polyline or Region for simple objects, and MultiPoint, MultiPolyline and MultiRegion for collections of objects),
- γ: alphanumeric values $\gamma = \gamma_1, \gamma_2, \ldots, \gamma_n$ accessed through the set of object's attributes $a_1, a_2, \ldots, a_n$ (for instance, the name of a street).

LoD layer and LoD object
LoD layers of a layer *l* correspond to the definitions of *l*'s objects in the scale ranges where they exist. A layer *l* can be seen as a serie of *n* LoD layers. LoD objects included in LoD layers can be matched (i.e. linked) between the two or more consecutive levels where they are represented. The matching configuration corresponds to the number of matched LoD representations of the same real world entities (when objects are represented at two different LoDs).

Three different matching cases are distinguished in Ai et al. (2001): *1:1*, *1:n* and *n:m* matching case. In our works only the *1:1* and *1:0* matching cases have been considered, i.e. the cases where 1 LoD representation of an object is mapped to 1 representation of the same object at a different and consecutive LoD or where it is not linked to other object (because it has been deleted between the two LoDs). Only these matching configurations are studied because use of incremental operators seems only relevant for these cases: more complex matching configurations should involve more complex increments which are less interesting.

The linking is based on identifier *o* of the object's different representations. Some solutions to match objects with various link cardinalities have been studied in Hampe et al. (2003).

A LoD representation or LoD object is an object *o* version defined for a level *i* in adequacy with a scale interval. It will be noted o^i .

The below described concept of increment is applied to polylines but can also be valid for regions. Indeed a region can be defined as a closed polyline: its boundary.

A polyline noted *P* is defined as a sequence of vertices $\{V_1, \ldots, V_n\}$ such that each couple (V_1, V_{i+1}) defines a segment $[V_1, V_{i+1}]$.

As we deal with multiple representations of same polylines, we use the following definition: a vertex V_i^j is a vertex *V* at index *i* of a polyline P^j.

For example, we consider two LoDs of a polyline in Fig. 4.6: a detailed and a simplified one. We can notice that vertices of P^n with indexes 1 and 4, i.e. V_1^n and V_4^n have the same coordinates that vertices of P^{n+1} with indexes 1 and 2, i.e. V_1^{n+1} and V_2^{n+1} .

We define the vertices which have the same coordinates, i.e. are matched, in the two LoD representations P^n and P^{n+1} as shared (or matched) vertices.
The set of matched vertices is used during the creation of increment and reconstruction of the polyline.

Increment
An increment allows performing changes on LoD object o^n in order to rebuild its representation o^{n-1} or o^{n+1}.
An increment point corresponds to a couple (op_i, V_i^j) where a geometrical operator op_i is combined with a manipulated vertex V_i^j .

An increment is defined as an ordered list of increment points. The increment allowing transition from o^n to o^{n+1} *(resp.* o^{n-1}) will be noted $Inc(o^n, n \rightarrow n+1)$ (resp. $Inc(o^n, n \rightarrow n-1)$).

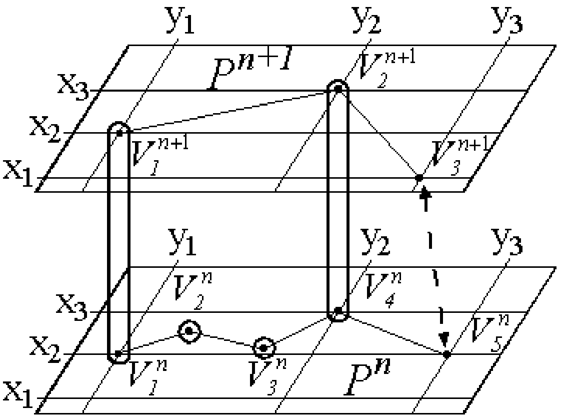


Fig. 4.6. Vertices of two LoD representations of a same polyline

Four geometrical operators are considered:

- *insert*: puts a vertex V which is only present in the most detailed polyline P^{n-1} at the index i of the less detailed one P^n during a transition from LoD n to LoD n - 1 (noted $n \rightarrow n-1$). It manipulates the index and coordinates of a vertex.
- *keep*: keeps a vertex V^n shared by P^n and P^{n+1} at index i of P^n,
- *remove*: removes a vertex V^n only present at index i of P^n,
- *move*: changes the coordinates of a vertex V in a polyline P^n. It manipulates index and coordinates shifts of a vertex for a given transition $n \rightarrow n-1$ or $n \rightarrow n+1$.

Geometrical operators *keep* and *remove* are used during a generalisation transition $n \rightarrow n+1$ and manipulate only the vertex index. The first one is used if vertices to keep are fewer than vertices to remove, and the second one if the number of vertices to keep is more important than those of vertices to remove (cf. section 4.4.3).

We use the following notations:

- ν_{insert} for the domain of inserted vertices defined as couples $\left(insert, V_i^{n-1}\right)$,
- ν_{keep} for those of kept vertices defined as couples $\left(keep, V_i^n\right)$,
- ν_{remove} for those of removed vertices defined as couples $\left(remove, V_i^n\right)$,
- and ν_{move} for those of moved vertices defined as couples $\left(move, V_i^n\right)$.

The domain of increments points is noted ν_{inc}, such that:

$$V_{inc} = \{V_{insert}, V_{keep}, V_{remove}, V_{move}\}$$

For a transition $m \rightarrow n$ of an object *o* the expression of an increment is:

$$Inc(o, m \rightarrow n) = \{o, (op_1, V_a^{lod}), \ldots, (op_i, V_b^{lod}), \ldots (op_k, V_c^{lod})\}$$

where *lod=m* or *lod=n* (manipulated vertex can belong either to o^m or to o^n) and $a < b < c$.

Refinement increments $Inc(o, n \rightarrow n-1)$ are constituted of increments points from V_{insert} and V_{move}, and generalisation increments $Inc(o, n \rightarrow n+1)$ are composed of vertices from V_{keep}, V_{remove} and V_{move}.

Each increment point (op_i, V_i^j) has an encoding cost (or size) C_{op_i} representing the number of bytes used to encode a vertex and an operation. This cost depends on the vertex part manipulated by the geometrical operator (index only or both index and coordinates). The total encoding cost C_{Inc} of an increment corresponds to the sum of costs of all its increment points. Thus it is an indicator of the time it will take to transfer the increment from a server to the client:

$$C_{Inc} = \sum_{i=1}^{k} C_{op_i}$$

In certain situations, representation of an object is not modified between two LoD: for example, if DP algorithm is applied with a low tolerance value. Set of couples $\{(op_i, V_i^j)\}$ is then replaced with an operator *nop* for these identical objects called O_{id}. *nop* operator is used for marking object *o* that must be kept without modifications. An increment $\{o, nop\}$ with $o \in O_{id}$ is used only during a generalisation transition because all objects of destination set (the more generalised one) come from the origine set (the more detailed one). This is a consequence of the selection/elimination process (cf. section 4.2.3).

Examples of increments

The generalisation and refinement increments of the cases illustrated in Fig. 4.7 correspond to the following sequences:

In the case A (where the number of shared vertices is greater than the number of inserted ones):

$$Inc(P, 1 \rightarrow 2) = \{(move, V_2^1), (remove, V_3^1)\}$$

$$Inc(P, 2 \rightarrow 1) = \{(move, V_2^2), (insert, V_3^1)\}$$

In the case B (where there are more inserted vertices than shared ones):

$$Inc(P, 1 \rightarrow 2) = \{(keep, V_1^1), (move, V_2^1)\}$$

$$Inc(P, 2 \rightarrow 1) = \{(move, V_2^2), (insert, V_3^1), (insert, V_4^1)\}$$

Increments act in a similar manner as EGO's defined in Sester et al. (2004b). But geometrical operators are used in order to reduce the amount of exchanged data by reusing LoD representations of objects available in client's cache and not to achieve a continuous generalisation like the EGO's. Furthermore in our case, the process of increment creation is independent of the generalisation. In this way, increments can be computed from data coming from different sources. The above described different concepts are used during client-server transfer of data.

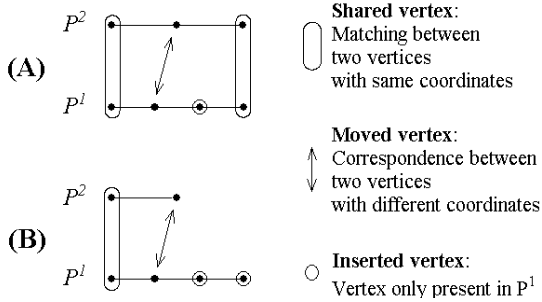


Fig. 4.7. Different configurations of polylines' LoD representations in 1-dimensional space

4.3.2 Transfer and management principles

Types of multi-resolution data

Different types of data implied in an embedded navigation application for visualizing MR data have been distinguished in Follin et al. (2005a). We have considered the real-time navigation of a mobile user across two LoD representations of a same thematic layer where the user requests are based on both its location and the zoom level. Only data relevant to it location are downloaded.

Three types of requested data have been identified depending on locally available LoD data:

- already available dataset that can be reused from the same level of detail,
- dataset that can only be reused from the previous level of detail, called useful objects O_{util},
- dataset that is unavailable on the client for all LoD layers and needs to be retrieved from the server. It can be either objects omitted during generalisation process (in the case of a refinement transition), or newly displayed objects (in all cases).

Transfer models

In Stockus et al. (2001), three schemes of data transfer between client and server have been defined in order to reduce the volume of exchanged data:

1. The simple communication mode where, upon a query of the client, the server will compute and send the complete answer.

2. The two-step communication schema where all queries are still executed on the server but the client maintains data cache and can reuse already received objects ;
3. The pre-computed answer mode where the client can execute some queries locally without connection to the server.

In Follin et al. (2005a) we have proposed a general transfer model of MR data based on the communication with a pre-computed answer. MR data transfer is performed in order to reuse data already locally available at a LoD different from requested one. Thus, local processing of LoD objects can be made as answer to a client request of transition between a LoD m to a LoD n representation (zoom in or out), i.e. from a source layer to a destination one. It can also be made each time that required data are covered by data available at a different LoD (during a pan operation or as a consequence of the user's displacement).

This data transfer model is decomposed in four steps:

1. Local computing of the identifiers of objects reusable at the destination and source layers,
2. Sending of a request to the server including identifiers of the destination and source layers, completed by two sets of identifiers: objects available at queried level n and objects O_{util}^{m} exclusively available at source level m,
3. Sending by the server of an answer which mainly includes (if there is no data update) missing objects at both LoDs and increments set $Inc\left(O_{util}^{m}, m \rightarrow n\right)$ allowing reuse of objects O_{util}^{m} only available in level m and required for level n,
4. Rebuilding of missing LoD n objects from same objects O_{util}^{m} available at level m and $Inc\left(O_{util}^{m}, m \rightarrow n\right)$.

Comparison between mono-resolution and multi-resolution strategy

This transfer strategy can be called multi-resolution strategy because it is based on incremental reconstruction of data: not only available object at level n but also useful objects of level m (i.e. data only available in l^m and reusable for l^n) are considered on the client side. This strategy can be compared with a mono-resolution one for which only objects available at required level n are taken into account. Efficiency of our model for reducing the amount of transferred data can be evaluated with such a comparison (section 4.5.4).

4.4 Incremental strategy: conditions and interest

4.4.1 Discussion about increment creation and reconstruction

Two types of functions are distinguished in incremental strategy: those for creating increments from two LoD representations of the same object and those for reconstructing the LoD representation of an object from another LoD of the same object and the corresponding increment.

The first are performed on the server side and the second on the client side. More details on formal functions and algorithms are given in Follin et al. (2005b). The reconstruction algorithms present a linear complexity because only one browsing of increment is necessary for reconstructing a polyline. So it can be performed rapidly on the client side where computing resources are limited. By consequent, it will be supposed that the gain in transmission remains interesting in spite of the client side process of increment reconstruction.

4.4.2 Required conditions

If LoD n objects set O^n is required on the client-side then transfer from the server and use of an increment set $Inc(O^m, m \rightarrow n)$ rather than transfer of O^n depends on the following conditions described in Follin et al. (2005b):

- existence of a set O_{app} of objects matched between O^m and O^n,
- for each matched object of O_{app}, existence of a set of shared vertices,
- ratio on the sizes of the different LoD representations geometries (more detailed object must contain more vertices than more generalised one)
- from the transfer point of view, a significant reduction of the size of $Inc(O^m, m \rightarrow n)$ compared with the size of O^n.

The first three conditions are “structural”: data have to verify them. The fourth one is linked to the modeling and encoding of increments and objects. By computing the costs C_{Inc} of increments, we can consider the cases where transfer and use of an increment is more interesting than transfer of the corresponding LoD object.

4.4.3 Cost of increments and efficient objects

The theoretic costs of different increments points can be established by taking into consideration a specific encoding of data used by different geometrical operators. We consider sizes of Java primitive types to evaluate the cases where it is more interesting to use increment rather than “entire” objects:

- Each couple of coordinates (*x, y*) are encoded by two double on 2 × 8 bytes,
- Each index *i* is encoded by an integer on 4 bytes

Geometrical operators can work with the same vertices parts: *move* and *insert* use both coordinates and index, *keep* and *remove* use an index. In order to determine the operator to apply, from the implementation point of view, a variable *op* encoded by one byte is used. This encoding is not "perfect" but can be considered as a way to implement our concepts in a simple data structure and to measure its efficiency.

In addition to increment points of v_{inc} we can define the "entire" vertices of v: they correspond to couples $\left(download, V_i^n\right)$ where operator *download* manipulates coordinates *x* and *y* of each vertex *V* composing an object o^n. This operator is used to measure efficiency of multi-resolution strategy in comparison with a mono-resolution one (Fig. 4.8).

From the transfer point of view, we can measure interest of downloading increments points rather than LoD representation by taking into account proportion between different categories of points. We consider two LoD polylines P^1 and P^2: vertices composing them can be inserted, shared or moved.

Vertices' numbers of polylines P^2 and P^1 are respectively noted *S2* and *S1*:

- P^2 is composed of n_{shared} shared vertices and n_{moved} moved ones: $S2 = n_{shared} + n_{moved}$,
- P^1 is composed of n_{shared} shared vertices, n_{moved} moved ones, and $n_{inserted}$ inserted ones: $S1 = n_{shared} + n_{moved} + n_{inserted}$.

These notations of points numbers are used in association with cost notations in order to evaluate interest of our strategy during a generalisation transition, and during a refinement one.

Increments points	Cost notation	Cost (used data)	Cost (estimated in bytes)
v_{insert}	C_{insert}	x, y, i, op	21
v_{move}	C_{move}	$\Delta x, \Delta y, i, op$	21
v_{keep}	C_{keep}	i, op	5
v_{remove}	C_{remove}	i, op	5
v ("entire" vertex)	C_{dwnld}	x, y	16

Fig. 4.8. Costs of operators used in transformation between different LoD representations of a polyline

For a generalisation transition from P^1 to P^2 (i.e. for decreasing resolution) two strategies are possible:

- Download and use increments points V_{keep} of v_{keep} or V_{remove} of v_{remove} (conservation of shared vertices or removal of additional ones) and of V_{move} of v_{move} (displacement) in order to reuse P^1.
- Download *S2* vertices from v of P^2.

Keep operator is used if number of additional vertices is greater than those of shared ones, i.e. if $n_{inserted} > n_{shared}$. At the opposite, *remove* operator is more interesting if $n_{shared} > n_{inserted}$. Furthermore cost of different incremental operations must be smaller than those of downloading "entire" vertices.

If $n_{shared} < n_{inserted}$, the following equation needs to be respected:

$$C_{dwnld}(n_{shared} + n_{moved}) > C_{move}(n_{moved}) + C_{keep}(n_{shared})$$

It means we must have the following proportion between the polyline's vertices:

$$2{,}2 \times n_{shared} < n_{moved}$$

If $n_{inserted} < n_{shared}$, the following equation needs to be solved:

$$C_{dwnld}(n_{shared} + n_{moved}) > C_{move}(n_{moved}) + C_{remove}(n_{inserted})$$

It means that we must observe the following repartition of the polyline's vertices:

$$3{,}2 \times n_{shared} < n_{moved} + n_{inserted}$$

For a refinement transition from P^2 to P^1 (i.e. for increasing resolution) two strategies are considered:

- Download and use increments points V_{insert} of v_{insert} and V_{move} of v_{move} in order to reuse P^2.
- Download *S1* vertices V of P^1.

Incremental operations of insertion and displacement are more interesting if their global cost $C_{insert}(n_{inserted}) + C_{move}(n_{moved})$ is smaller than those of downloading polyline P^1, that is noted $C_{dwnld}(n_{shared} + n_{moved} + n_{inserted})$. It implies the respect of the same repartition between polyline's vertices than in the case of generalisation transition with suppression (when $n_{inserted} < n_{shared}$), i.e. $3{,}2 \times n_{shared} < n_{moved} + n_{inserted}$

These equations are used during increment creation on the server side: increments are only computed for objects for which points proportions between n_{shared}, n_{moved} and $n_{inserted}$ respect the conditions of efficiency given by them. These objects are noted O_{eff}.

Increment sets are computed between two datasets at consecutive levels of detail in a generalisation direction and in a refinement one. They are stored on the server side, recomputed if data are updated and transferred to a client for reusing data available in its cache. The reconstruction is finally performed on the client side to create the instances of objects by reusing the available LoD representations.

4.5 Implementation and results

These concepts have been experimented by simulating the navigation (with both pan and zoom operations) of a mobile user in the city of La Rochelle.
Three LoD layers representing the road network of La Rochelle have been produced by generalisation in order to reduce the data volume while preserving topological consistency on the one hand and to be adapted to an incremental management of data on the other hand (cf. section 4.2.3).

4.5.1 Constitution of datasets (generalisation and matching)

Source dataset correspond to IGN[2]'s Georoute® road data covering a zone of 13,7 × 15,53 kilometers around La Rochelle.
Original road objects contain on average 3,68 vertices. On such object simplification operators like DP algorithm are not interesting because only 1,68 vertices can be removed on average if we do not take into account extremities which are not eliminated. Consequently it does not potentialy represent an important "stock" of increment points.

The connected roads sections have been merged by considering their name attribute. In fact a street with the same name should be regarded as an integral part, for instance during a selection process. Moreover, it allows maximizing the number of increments points. After this step, polylines contained an average of around 8 vertices and were more adapted to an incremental strategy. This dataset corresponds to level *lod1* from which levels *lod2* and *lod3* have been derived by selection and simplification.

After a computing of unique identifiers for each *lod1* road objects, polylines have been selected according to their importance while conserving their identifiers. Selected objects of *lod2* and *lod3* layer (which can be seen in Fig. 4.9) have then been simplified with (cf. section 4.2.3):

- a contraction step on traffic circles,
- a second step of simplification with a modified version of DP algorithm allowing the preservation of topological consistency by conserving connection points between poylines. Tolerance values (3 meters for *lod2* layer and 30 meters for *lod3* one) have been chosen in order to obtain a maximal number of increments vertices. With such values, number of vertices of the matched objects between *lod1* and *lod2* layers has been reduced in similar proportions than between *lod2* and *lod3* layers.

[2] Institut Géographique National

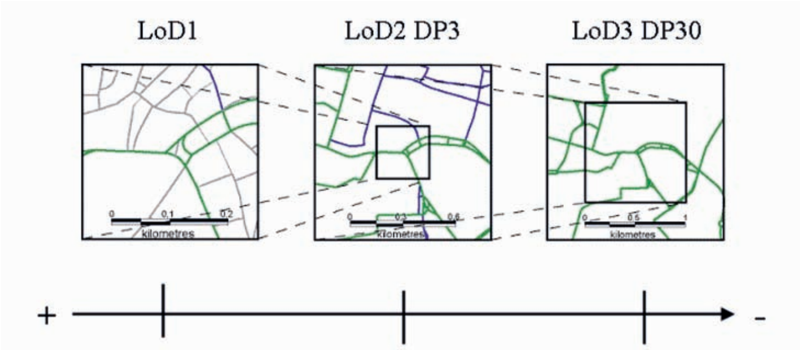


Fig. 4.9. Zoom out operation through 3 LoD layers of La Rochelle's transportation network

4.5.2 Dataset adaptability to our incremental strategy

In this section, we present indicators related to adaptation of our data to an incremental strategy. Dataset O_{eff} corresponds to a subset of matched objects of O_{app} which satisfy conditions of efficiency. We note $O_{\overline{eff}}$ the matched objects for which increment is not interesting, and $O_{\overline{app}}$ the unmatched objects. Whereas objects of O^{m}_{util} are those considered as "exploitable" from the client point of view, O_{eff} objects are those for which exist an increment on the server side.

Two indicators about increment efficiency have been computed between the three LoD layers noted l^1, l^2 and l^3:

- one proportion indicator, the ratio between the number $\left|O_{eff}\right|$ of polylines reusable of a layer l and the total number $\left|l\right|$ of objects in this layer: $\frac{\left|O_{eff}\right|}{\left|l\right|}$
- and a gain indicator which compares the theoretical cost of increment $Inc\left(O^{m}_{eff}, m \rightarrow n\right)$ with those of efficient O^{n}_{eff} objects of layer l^n, noted $100 - \frac{100 \times c\left(Inc\left(O^{m}_{eff}, m \rightarrow n\right)\right)}{c\left(O^{n}_{eff}\right)}$

The first indicator allows comparison of objects implied in incremental strategy with entire dataset and the second one is used to measure the interest of incremental strategy for a final user i.e. the diminution of sizes of transferred objects.

By using gain indicator we can compute the gain to reuse an entire layer in order to get a more detailed version or a more generalised one of the same layer. This gain is called “global gain” because it is made “globally”, i.e. without simulating a moving user (as in section 4.5.4).

4.5.3 Evaluation with “global gain” indicators

In Fig. 4.10 global gain is represented for our three LoD layers. Two cases are distinguished: in the first one (upper cubes) only “efficient” objects are considered and in the second one (lower cubes) all objects are taken into account. For the second case, the basic idea is that we dispose of a layer we want to reuse with increment rather than to download entirely the same layer at a different resolution.

Around 15 % of the layer l^1 objects are implied in increments with layer l^2. They are in correspondence with around 90 % of the l^2 objects.If we consider only O_{eff} dataset for which increments are efficient, observed gains vary from 25 to 30 % in the refinement direction and from 70 to 75 % in the generalisation one. If we take into account all the polylines, gain is of the same order for a generalisation transition (from 70 to 80 %) because no new objects are required. But the gain is less important in a refinement direction (8 % for $2 \rightarrow 1$ transition and 15 % for $3 \rightarrow 2$ direction). It is explained by the fact that entire objects must be downloaded: the unmatched ones.

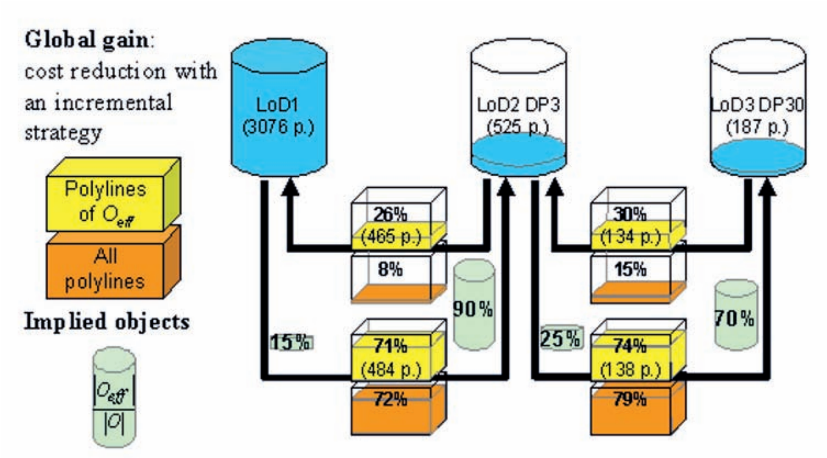


Fig. 4.10. Global gain with an incremental strategy

We can then conclude that importance of the gain mainly depends on the proportion of shared and reusable objects between the sets of data at different resolutions. Secondly it is linked to the cost of increment points and to the number of O_{id} identical objects.

4.5.4 Evaluation with "scenario-oriented" simulations

Data and principles of simulation

The objects sets implied in the mono-resolution and multi-resolution strategies have been computed through a simulation in order to measure interest of our strategy in "real" conditions. Experimentations have been made by simulating navigation of a mobile user in La Rochelle city with three GPS routes, i.e. three sets of coordinates collected at regular time steps with a GPS equipped car (Fig. 4.11). Volumes of data exchanged between client and server have been computed in the cases of strategies with and without reuse (cf. section 4.3.2). Navigation through different resolutions has been simulated while varying the number of zoom operations from 5 to 30.

Fig.4.11. one of the GPS routes (in red) used for the simulation of navigation in the streets of La Rochelle (in black)

Results

The data sent by the client and transferred from the server have been computed for MR strategy and mono-resolution one. Results are presented in Fig. 4.12. Size reduction of transferred objects appeared as globally satisfactory. It is more important between l^2 and l^3 than between l^1 and l^2: it is clearly linked to the fact that proportion of "efficient" objects is larger between these LoD layers.

Number of zooms	Between l^1 and l^2	Between l^2 and l^3
30	5,11	15,80
20	4,89	16,01

10	5,43	15,96
5	5,71	15,65

Fig. 4.12. Average percent of size reduction of transferred objects between consecutive LoDs

4.6 Conclusion and outlook

In this chapter a complete solution for the production and the management of multi-resolution geodata in a mobile system is proposed. It is validated through a global gain and a transfer simulation. Our approach is based on a data model where concepts of layer and object are extended to different LoD representations. The core of this framework is the concept of increment: it allows performing changes on an object representation at some LoD in order to rebuild its geometry at a different consecutive LoD. It can be adapted to the required direction of transition (generalisation or refinement) by using only the useful part of manipulated vertices. In this way it allows reduction of data transfer between server and client each time that requested data are covered by data available at a different LoD.

We plan to implement a solution based on a new formalism allowing computation of "optimal" increment with a minimal cost from two LoD representations.
Furthermore other increments could be defined: for instance in Brenner et al. (2003), operator *InsertInEdge* appears interesting with its low cost.
Finaly, our strategy could be applied to LoD datasets coming from different sources: matching operators should be defined in order to make such datasets respecting our conditions.

Acknowledgements

This research was partially funded by the Communauté d'Agglomération of La Rochelle, France.

References

Ai, T., van Oosterom, P. (2001): A map generalisation model based on algebra mapping transformation. In: Proceedings of the Ninth ACM International Symposium on Advances in geographic information systems, Atlanta (USA), ACM Press, pp. 21–27.

Bertolotto, M. (1998): Geometric Modeling of Spatial Entities at Multiple Levels of Resolution. PhD Thesis in Computer Science, Università degli Studi di Genova.

Bertolotto, M. and Egenhofer, M. J. (2001): Progressive transmission of vector map data over the World Wild Web. *GeoInformatica*, Vol. 5, No 4, pp. 345-373.

Brenner, C. and M. Sester (2003): Continuous Generalization for Small Mobile Displays. *International Conference on Next Generation Geospatial Information*, Boston (USA), October 19-21, 2003

Buttenfield, B. (2002): Transmitting Vector Geospatial Data across the Internet. Proceedings of the Second International Conference on Geographic Information Science. Boulder, Colorado (USA), Eds. Springer-Verlag pp. 51-64

Buttenfield, B. (1999): Progressive transmission of vector data on the internet : A cartographic solution. *Proceedings 18th International Cartographic Conference*, Ottawa (Canada).

Cecconi, A. and Galanda, M. (2002): Adaptive Zooming in Web Cartography. *Proceedings of SVG Open 2002*. Zurich (Switzerland), pp. 78-83.

Cecconi, A. (2003): Integration of cartographic generalization and multi-scale databases for enhanced web mapping. PhD Thesis, University of Zurich.

Follin, J.M., A. Bouju, Bertrand, F. and P. Boursier, (2005): Multi-resolution extension for transmission of geodata in a mobile context. *Computers & Geosciences* 31, pp. 179–188

Follin, J.-M., Bouju, A., Bertrand, F. and A. Stockus (2005): An Increment Based Model for Multi-resolution Geodata Management in a Mobile System. *Proceedings of the 5th International Workshop W2GIS 2005,* Lausanne (Switzerland), December 15-16, Lecture Notes in Computer Science, Vol. 3833, Springler Verlag, pp.42-53.

Hampe, M., Anders, K.-H. and M. Sester (2003): MRDB applications for data revision and real-time generalisation. *Proceedings of 21st International Cartographic Conference*, Durban, South Africa, 10 – 16 August 2003, pp. 192–202.

Hoppe, H. (1996): Progressive Meshes. *Proceedings of SIGGRAPH 96*. Août. Eds ACM SIGGRAPH, Annual Conference Series, New Orleans (USA). pp. 99-108.

Jones, C.B., Abdelmoty, A.I., Lonergan, M.E., van der Poorten, P. and S. Zhou (2000). Multi-scale spatial database design for online generalization. In Pip Forer et al., editor, *9th International Symposium on Spatial Data Handling (SDH 2000)*, Beijing (Chine), Août 2000. International Geographical Union, pp. 7b.34–44.

Lehto, L. and Kilpeläinen, T. (2001): Real-time generalization of XML-encoded spatial data on the web. In D. B. Kidner and G. Higgs, editors, *Proceedings of the GIS research UK, 9th Annual Conference GISRUK 2001*, University of Glamorgan, Wales (United Kingdom), April 2001, pp. 182–184.

Oh, Y.-H. and Bae, H.-Y. (1999): Advanced progressive transmission for spatial data in web based GIS. *Second AEARU Workshop on Web Technology*, Kaist, Taejon (South Corea), pp. 41–46.

Persson, J. (2004): Streaming of compressed multi-resolution geographic vector data. *Proceedings of 12th International Conference on Geoinformatics - Geospatial Information Research : Bridging the Pacific and Atlantic*, Gaevle (Sweden). Eds S. Anders Brandt. Gavle University Press, pp. 765–772.

Saalfeld, A. (1999): Topologically consistent line simplification with the douglas-peucker algorithm. Cartography and Geographic Information Science, Vol. 26 No.1, pp. 7–18.

Sester, M., Sarjakoski, T., Harrie, L., Hampe, M., Koivula, T., Sarjakoski, T., Letho, L., Elias, B., Nivala, A.-M. and Stigmar, H (2004): Real-time generalization and multiple representation in the GiMoDig mobile service. Public report D7.1.1-D7.2.1, D7.3.1, GiMoDig-project. http://gimodig.fgi.fi/deliverables.php (accessed April 2007).

Sester, M., Brenner, C. (2004): Continuous generalization for fast and smooth visualization on small displays. *International Archives of Photogrammetry and Remote Sensing*, Vol. 34, Istanbul (Turkey), ISPRS

Stockus, A., Bouju, A., Bertrand, F. and P. Boursier (2001). Accessing to spatial data in mobile environment. In *Proceedings of 2nd International Conference on Web Information system*, Kyoto (Japan), December 2001, pp. 414–422.

Tomlinson, R. (1967): An Introduction to the Geographic Information System of the Canada Land Inventory. Departement of Forestry and Rural Development, Ottawa, Canada

van Kreveld, M. (2001): Smooth generalization for continuous zooming. *Proceedings of 20th International Cartographic Conference*, Beijing (China), pp. 2180–2185.

van Oosterom, P. (1995): GIS and Generalization Methodology and practice, chapter The GAP-Tree, an approach to 'on-the-fly' map generalization of an area partitioning. GISDATA1. Taylor & Francis, european science foundation edition. pp. 120–132.

Yang, B. (2005): A multi-resolution model of vector map data for rapid transmission over the Internet. In *Computers & Geosciences,* Vol. 31, Issue 5, June 2005, pp. 569-578

Zhou, M. and Bertolotto, M. (2005): Efficiently generating multiple representations for web mapping. . *Proceedings of the 5th International Workshop W2GIS 2005,* Lausanne (Switzerland), December 15-16, Lecture Notes in Computer Science, Vol. 3833, , Springler Verlag, pp.54-65.

5 Evaluating the Effectiveness of Non-Realistic 3D Maps for Navigation with Mobile Devices

Malisa Ana PLESA, William CARTWRIGHT

School of Mathematical and Geospatial Science, RMIT University, Melbourne, Victoria, Australia

Abstract. Small mobile computer platforms are being employed to deliver maps and map-related objects to users, 'at location', on-demand and almost instantaneously. The products delivered are mainly conventional in design, sometimes only mimicking their paper counterparts. However, a number of applications have introduced innovative presentations as both 2D and 3D images. The delivery of 3D images on these devices, particularly as realistic impressions has been the focus of recent research to evaluate the effectiveness of such images for navigation.

This chapter provides a background on the use of 3D imagery by cartography. It then describes the initial stages of a project that built 3D images for mobile de-vices based on Döllner's theory related to non-realistic 3D images. The research applied Döllner's theory to the realization of non-realistic 3D images for PDAs. It then outlines the development of a 'proof-of-concept' prototype and it pro-vides the results of an evaluation of this prototype. Finally it discusses possible applications of such imagery.

5.1 Introduction

Currently, there is much interest in the creation and display of photorealistic imagery on mobile devices, but no evidence exists to suggest that it is the most appropriate method to convey spatial information. Photorealistic 3D maps designed for display on small screen devices must cope with costs associated with the development of realistic imagery as well as the restricted processing and display capabilities of these devices. This type of representation may also lack in the areas of user acceptance and understanding. Non-photorealistic rendering is a new revelation in computer graphics that aims at suppressing detail whilst emphasising important features. This chapter reports on research undertaken to evaluate the potential of non-photorealistic computer graphics for the display of 3D city maps on mobile devices.

The chapter begins with an overview of computer graphics and photographic realism and provides examples of non-photorealistic rendering. It then addresses photorealism vs. non-photorealism. This is followed by a section that focusses on how cartography has employed 3D and provides historical and contemporary examples to illustrate this section. The focus then moves to mobile maps and the design considerations for maps on small, mobile devices. The next section outlines the concept of expressive city models for small-screen delivery. It then outlines a research project that evaluated the use of non-realistic 3D models on small-screen de-vices and their use as navigation aids. The results from this research are provided and areas for potential future research are outlined.

5.2 Computer graphics and photorealism

Since the introduction of computer graphics, the ultimate goal was to achieve photographic realism (Schumann et al., 1996; Durand, 2002; Gooch and Gooch, 2001). Following the introduction of the first computer aided drawing system, DAC-1 in 1959, computer technology has improved to allow for the generation of high quality photorealistic imagery. The value of an image was often judged by how closely it resembled reality, and today, these graphics can often be indistinguishable from photographs. The creation of this type of imagery requires a very high level of detail, even if this results in a cluttered and confused composition (Gooch and Gooch, 2001).

5.2.1 Is photorealism the only answer?

There is no clear evidence to suggest that photorealism is the most effective method of presenting visual information, and little research has attempted to explore alternative methods of information display (Schumann et al., 1996; Markosian et al., 1997; Ferwerda, 2003; Gooch and Gooch, 2001). It has been assumed that humans have the ability to understand realistic imagery because they are familiar with how reality 'looks' (Collinson, 1997). A problem that needs to be addressed is that many applications may not require photorealism. What good is a photograph when you are physically within the scene and can see the information for yourself? Perhaps a different image, one that supports location awareness and navigation, would be better.

How realistic is a particular image? Is it possible to measure realism? These questions remain fuzzy and relatively unanswered in the field of computer graphics (Durand, 2002). In her book, Varieties of Realism (1986), Hagen discusses the concept of different varieties of realism achieved through different methods of artistic illustration. Ferwerda (2003) supported this idea and discovered that it could also be applied to digital imagery. He went on to define three varieties of realism in computer graphics:

- Physical realism – providing the same *visual stimulation* in the image as that received from the original scene;
- Photorealism – providing the same *visual response* in the image as that exhibited by the original scene; and
- Functional realism – providing the same *visual information* in the image as that gained from the original scene.

Both physical realism and photorealism require enormous amounts of data to achieve the desired result, and their creation is expensive and time consuming. Large file sizes often cause this type of imagery to be too slow for interactive applications. While photorealism provides a delineation that is visually correct, it is important to ensure that the desired information is communicated in a functional context (Gooch and Willemsen, 2002). Ferwerda (2003) uses the term ‘functional realism’ to describe digital imagery that provides useful knowledge about the properties of objects, allowing users to make reliable visual judgements. It is functional because the same information is communicated to all users. This contrasts with physical realism and photorealism, because the information that users extract from these renditions differs according to their personal preferences and understanding. By no means does this suggest that functional realism departs completely from reality. While this type of imagery may not appear to be visually real, it is functionally real as it allows users to successfully perform real world tasks (Ferwerda, 2003).

5.2.2 Non-photorealistic rendering

Knowledge and techniques long used by artists are now being applied to computer graphics to emphasise specific features, expose subtle attributes, and omit extraneous information (Gooch and Gooch, 2001). Non-photorealism is a pictorial style that represents a form of functional realism. Currently, the term ‘non-photorealistic’ lacks a clear definition. Durand (2002, p.112), states that “The only meaning of non-photorealistic is that the picture does not attempt to imitate photography and to reach optical accuracy”. While generally being accepted as the opposite of photorealism, non-photorealism tends to adopt a different meaning referring to a different kind of realism, depending on the field of research (Konig et al., 2000).

Non-Photorealistic Rendering (NPR) is a rapidly growing area of interest in computer graphics (Schumann *et al.*, 1996; Markosian *et al.*, 1997; Goldstein, 1999; Herman and Duke, 2001; Halper *et al.*, 2002; Döllner and Walther, 2003). Its aim is to develop algorithms to allow for the generation of abstract imagery, which work to emphasise important features while suppressing unimportant details. These methods refer to any image processing systems that simulate specific artistic techniques or more generally, styles that do not resemble photographs (Mignotte, 2003). At the moment, the technique cannot be entirely automated, and requires user input to control the parameters for a certain rendering style. Non-photorealistic rendering techniques rarely come up with ‘new’ styles, but tend to emulate non-digital artistic techniques, such as ink painting, charcoal drawing, or engraving.

Currently, NPR is being used for a variety of purposes, including medical textbooks (Gooch & Gooch, 2001), architecture (Schumann et al 1996), applications with new interaction methods (such as haptic devices) (Herman & Duke, 2001) and for the communication of 3D structure (Finkelstein and Markosian, 2003). Its ability to highlight crucial features, whilst suppressing unnecessary detail has given NPR a significant advantage over traditional photorealistic methods of rendering.

5.2.3 Photorealism vs. non-photorealism

Whilst not possessing optical accuracy, a non-photorealistic image is often clearer than a photograph. This is because it can omit redundant elements and maintain those that are relevant (Gooch and Willemsen, 2002; Lum and Ma, 2002; Gooch and

Gooch, 2001). In many situations, presenting an observer with enough information to create the illusion of reality is often more important than simulating reality. While photorealism leaves nothing to the imagination, abstract imagery can often be more effective with communicating subtle information, capturing relationships and highlighting crucial features.

Some researchers have questioned the need for realism in many graphics applications (Feiner et al., 1988; Gershon et al., 1996; Schumann et al., 1996; Herman and Duke, 2001; Durand, 2002; Ferwerda, 2003; Gooch and Gooch, 2001). An image is not the same as the object it is illustrating, and a visual depiction can 're-present' selected properties of the original scene according to user requirements (Ferwerda, 2003). Non-photorealistic graphics should not be seen as a competitor to photorealistic graphics (Herman and Duke, 2001). There are many situations where non-photorealism can be more effectively applied, although circumstances that call for optimal realism are abundant. For example, non-photorealistic imagery has been used in medical textbooks to illustrate structure, but cannot replace a photograph whose purpose is to illustrate a specific skin condition.

5.2.3.1 Time and cost considerations

The highest level of detail is generally preferred in photorealistic graphics. This makes detail very hard to neglect, and causes problems during the data collection, creation and delivery stages of image production (Gooch and Gooch, 2001). Realism is expensive. The cost is due to the vast amount of detail required, time spent on image production, and expertise required of image developers. Photorealistic images can often be too slow for interactive applications, and a loss of image value occurs when complexity is reduced. This complexity also puts a strain on digital displays and does not cope with challenges imposed by growing internet usage. Image files are too detailed to be compressed acceptably, and today's screens have limited display capabilities (Herman and Duke, 2001). All of these problems escalate when dealing with 3D graphics because of the added intricacy.

Non-photorealistic graphics are effective in eliminating the above problems associated with photorealistic imagery. Time is decreased because precise detail is not required, nor desired, and output appearance can be controlled because it is not strictly limited to reality (Goldstein, 1999). This type of imagery compresses satisfactorily, so it can be effectively displayed on digital devices, while also being easily transferable over the Internet. Non-photorealistic images are also far simpler to create, and are being effectively applied to 3D graphics (Finkelstein and Markosian, 2003).

5.2.3.2 User understanding

To date, very little perceptual research has been directed at NPR images (Schumann et al., 1996; Gooch and Willemsen, 2002), but the field is starting to take steps that address cognitive theory (Herman and Duke, 2001). Studies have shown that human image interpretation is influenced by factors that have little to do with realism, and that the human mind is able to complete abstract information through the cognition process (Duke et al., 2003). Human understanding of the world is not based primarily on surface phenomena, but also involves deeper levels of representation that capture and reflect relationships and regularities at higher levels of abstraction (Duke et al., 2003). For example, Fig. 5.1 shows Kanizsa's Triangle, an optical illusion comprised of three sectored discs and some lines. Humans can perceive an upright equilateral triangle, seemingly above the other pattern elements even though it does not exist.

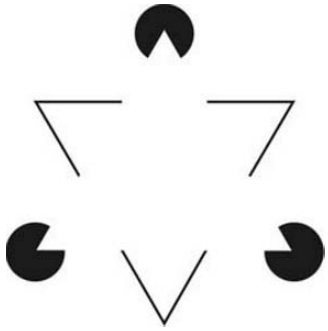


Fig. 5.1. Kanizsa's Triangle.

5.3 3D and cartography

Cartographers have always been interested in the mapping of the third dimension. This can be witnessed throughout history in town plans, bird's-eye views and relief representation. The weakest point of the traditional two-dimensional map is its representation of reality. All physical features that exist on the map in plan view, exist in three-dimensions in reality (Keates, 1989), and humans have a natural tendency to visualise spatial information in profile rather than as flat maps (Patterson, 1999). It is estimated that at least a third of the brain is involved in vision, and that 3D representations stimulate more neurons (Swanson, 1999). This causes a large portion of the brain to be involved in the problem solving process. It is believed that 3D maps may be more understandable to novice map users because they offer visualisation advantages that cannot be provided by 2D maps.

Two-dimensional images can record space, but cannot capture spatiality (Swanson, 1999). Three-dimensional maps have the power to provide a sense of how things relate to each other in space. The vertical characteristic of physical features is very important because it is a part of the landscape character, and can serve for identification purposes. This instigates partiality towards 3D representations of urban areas, despite the difficulties involved in their production (Keates, 1989). There has also been some evidence suggesting that users are able to recognise landmarks and find route easier with a 3D model rather than a symbolic 2D map (Kray et al., 2003).

5.3.1 3D maps throughout history

The mapping of the third dimension has always posed problems (Raisz, 1948). Before accurate measurement was attainable the height attributes of features were more or less unknown. Early cartographers also encountered difficulties because they were unfamiliar with how the world appeared from above. The representation of relief is believed to be the first 3D attribute attempted. This was eventually followed by the portrayal of cities in 3D.

Even the oldest known maps, etched on clay tablets, attempted to show mountain ranges. An interesting aspect of these early methods of elevation portrayal is the way in that the mountains were drawn in profile whereas other items were presented in plan view (Hodgkiss, 1981). This abstract technique of depicting mountains in profile view continued for many centuries. Common forms of early cartographic mountain illustration were stylised 'humps' (Fig. 5.2), which bore no relationship to the heights they were representing (Hodgkiss, 1981). These early attempts at relief representation aimed for the most effective visual technique, rather than the most accurate portrayal (Robinson, c1995).

Fig. 5.2. Stylised 'humps' for mountain illustration. (Coronelli, 1693, Source: Hodgkiss, 1981, p.40)

City maps differ from road maps and topographic maps because they are generally presented at a larger scale. This allows for the addition of appropriate details required by travellers. The goal of 3D city maps is to convey spatial information about the urban scene while remaining clear and functional (Hodgkiss, 1981). Relief is rarely presented on maps of urban areas due to the dense nature of the information posed by many prominent features.

Early urban cartographers were often not only concerned with showing city street layouts, but also with depicting the architectural style associated with the city. This technique often sacrificed accuracy in favour of pictorial styles (Elliot, 1987). Early city maps are now considered works of art because of their imaginative qualities and ability to evoke an emotional response.

Pictorial bird's-eye views were used to show towns in the earliest urban maps. These represented the area from a high oblique angle, conveying vertical dimension and architectural features, whilst relying on perspective rather than scale. These types of representations were acceptable because early cities were walled and isolated. This called for large-scale maps, whose coverage did not need to extend beyond the city walls. Fig. 5.3 is a bird's-eye view of Venice (1547). The design was intended to focus mariners' attention on landmarks, rather than layout, to aid their navigation.

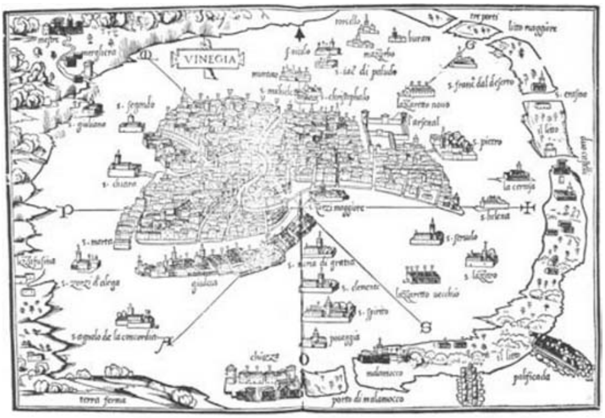


Fig. 5.3. Bird's-eye view of Venice (Bordone, 1547. Source: Hodgkiss, 1981, p.134).

The 1970s saw the introduction of computer-assisted systems aimed at automating cartographic drawing, and the terms computer-assisted cartography and 'digital map' were coined. These advances not only changed the process of mapping, but also the concept of mapping (Monmonier, 1982). Cartographers embraced the potential of the computer and used it to revolutionise cartography.

Another factor that modernised cartography in this era was the growing availability of accurate geographic data. The development of remote sensing around this time provided the cartographer with imagery that proved to be a reliable data source, also holding accurate height data for relief representation (Häberling and Hurni, 2002).

Computer technology today provides cartographers with a multitude of digital tools to aid in the map-making process. This has opened the door for 3D representations such as Virtual Reality (VR) systems, visualisations and city models. These techniques generally tend towards photorealistic depictions, which require a vast amount of base data and are still computationally expensive when compared to 2D computer graphics (Jones, 1997).

5.3.2 Is photorealism necessary?

By following the development of 3D maps throughout history one can see how the depiction of the third dimension has been altered to comply with advancements in technology and geographical knowledge. Now that technology has improved to permit the creation of high quality, 3D, photorealistic graphics, we need to consider its functional benefits. Are modern cartographers preoccupied with what technology can provide rather than what is fundamentally useable? In the case of city wayfinding, do users need to know every minute detail of a building's façade in order to navigate effectively? There has been very little research committed to identifying the benefits of realism in this context.

Is there a better way to communicate 3D information? Photorealism presents information as it is seen by the naked eye, but in reality there is a need for a technique that works to highlight the most important details, whilst eradicating unnecessary information; something that is readable rather than believable, because that is what map users do – they *read* maps. By reading maps, users are attempting to understand the 'mapped world', not the physical world, nor the map itself (Muehrcke et al., 2001). *"What makes a map so useful is its genius of omission"* (Muehrcke et al., 2001, p.11). Omission, which is frowned upon in photorealism, is the key to organising and presenting spatial information. It is a fundamental consequence of the generalisation process, and is the common factor present in all maps. After striving for photographic realism, the computer graphics community and other disciplines have come to discover the benefits of non-photorealism. It is possible that these benefits could have a positive impact on the field of cartography as well.

Maps are often known to be correct representations of geographic reality, although this is not always the case. Many maps are functional because they present non-realistic information. This is the case with strip maps, whose purpose is to distort geographic reality to simplify routing information. This was found to be an effective method of communicating information to road travellers, as extraneous detail was omitted to focus on the route itself. Similarities can be witnessed in some of today's route maps, including that of London's underground train network. The London Underground map was originally designed to be accurate in terms of both distance and direction, but this became confusing to travellers, as the train network grew more complicated. In 1933, electrical draughtsman, Harry Beck, presented a simplified representation of the underground network based on the circuit diagrams he drew for his job. All train routes were depicted as straight lines angled on increments of 45 degrees. The map used a limited number of colours, eliminating the need for a legend, whilst completely abandoning scale. Once released, the map was an instant success because it was clear and comprehensible. By comparing Beck's map to the original, it is easy to see the way in which geography has been distorted in favour of simplicity (Fig. 5.4). Beck's influence can be witnessed today in the design of many rail network maps worldwide.

Many 3D city maps depart from reality to optimise readability. Turgot's *Plan de Paris* (1734-9) is a remarkable map providing a historical record of Paris. It places emphasis on architecture through the use of a pictorial style, and utilises the shade of white to contrast city streets. In order to present the streets and building facades with minimal obstruction, street widths were exaggerated and buildings were 'moved' to achieve maximum visibility. A similar approach has been taken in modern times, with the Bollmann map series. Bollmann's map of Midtown Manhattan features exaggerated street widths, and is presented in an isometric projection to conserve visibility and scale. These maps are functional, providing an adequate amount of information to be used for many purposes.

Fig. 5.4. The original London Underground map (right) compared to Beck's 1933 map (Source: Black, 2003, p.134-5).

5.4 Mobile maps

The computer revolution has also provided the cartographer with a variety of digital output options. Society today is predominantly driven by technology, which has led to an increase in the rate of mobile device use (Christie et al., 2004). Falling prices have made this new technology accessible to a wider public. Cartographers have embraced this trend, and digital maps are no longer restricted to stationary computers. Mobile maps address the important requirements of portability and accessibility. These have never been completely fulfilled by the folded paper map, because it cannot be orientated in the direction of travel whilst featuring annotation the right way around (Wildbur, 1989).

5.4.1 User needs

Users of mobile technology seek accurate displays, but are primarily concerned with quick and easy accessibility (Wildbur, 1989). They are not interested in receiving the most information, but that which is the most relevant and condensed (Bieber and Giersich, 2001). Compared to stationary electronic devices, the use of mobile devices presents users with a higher cognitive load (Kray et al., 2003). This is due to the fact that they are most likely undertaking multiple tasks simultaneously. By conducting usability studies, optimal methods to reduce and simplify interaction can be identified.

5.4.2 3D maps on mobile devices

The interest in mapping the third dimension has also shifted towards maps designed for display on small screens (Vainio et al., 2002). Previous work has found that simple maps on mobile devices were limited in usefulness (Graham et al., 2003). Field trials and usability studies have identified advantages associated with the use of 3D graphics for navigation in urban areas (Rakkolainen and Vainio, 2001). To date, three-dimensional maps on small screens have also tended towards realistic representations. High-quality 3D maps on small devices face similar inconveniences relating to time and cost as those witnessed in the computer graphics domain. These also cause many problems when confronted with the limited processing power and display restrictions of these devices.

Research undertaken by Vainio et al. (2001, 2002) and Kray et al. (2003) focuses on photorealistic 3D city maps on mobile devices. Prototypes developed in both research programmes were designed to be navigational tools, so the findings were greatly concerned with human factors. Many problems associated with the creation and display of high quality realistic imagery were encountered, and the model had to be simplified in order to run smoothly on a handheld device (Vainio et al., 2002). However, a difficult trade-off was discovered when it was found that users believed a more detailed and realistic representation would have been more effective (Kray et al., 2003). This was a predictable outcome, as a vast amount of detail is required to make photorealistic imagery credible (Döllner and Walther, 2003).

Previous research attempts have been made to describe the ways in which realistic depictions can be utilised in 3D mobile mapping. It dismisses current creation and display problems by suggesting that they will lessen in the future when technology improves and costs decrease (Vainio et al., 2002). To date, there has been no investigation into the use of non-photorealistic imagery in place of photorealism for 3D mapping. There seems to be too much interest in the portrayal of information in a visually realistic manner, rather than a much-needed focus on discovering what is functionally realistic.

5.5 Expressive city models

In their paper "Real-Time Expressive Rendering of City Models", Döllner and Walther (2003) present a non-photorealistic rendering technique aimed at producing comprehensible assemblies of 3D urban objects. Their technique combines principles of cartography, cognition and non-photorealism, and has design techniques derived from methods of non-digital drawing. Expressive rendering holds a number of advantages over photorealistic representations. Unlike realistic 3D environments, less graphical and geometric detail is required to produce favourable results. The method also has use in many applications, such as architectural development, geographical information systems and transport information systems.

5.5.1 The rendering technique explored

Döllner and Walther (2003) outline a number of principles derived from non-digital techniques of drawing. These have been compiled from the perspectives of cartography, cognition, and non-photorealism. When combined, the principles below are believed to form the basis of effective abstract, 3D city portrayal:

(a) Geometric Projection: perspective views are arguably the most 'natural' way to present 3D information, however, the scale of objects is not consistent throughout the image, and larger objects in the foreground easily obscure important information (Keates, 1989; MacEachren, 1995). Orthogonal views are more easily constructed than perspective views, and they completely eliminate scale distortion. They provide a uniform relationship along all three axes, allowing for effective comparison between the heights and lengths of image objects (MacEachren, 1995).

(b) Graphical Techniques: colour is a major component of map design whose effective use enhances the perceptability on a map, clarifies and simplifies information, and distinguishes between groups of symbol categories. Certain colours are automatically recognized by map users as being representative of certain features; for example, blue is often used for water features and green to highlight vegetation. Black is generally reserved for the use of lettering and point symbols, but can provide high visual contrast when used sparingly for other important detail. The proposed technique adheres to colour conventions and uses black to create contrasting edges.

(c) Geometrical Techniques: all maps are abstractions of reality (Robinson, c1995). Map communication is enhanced through the processes of selection and generalization, which determine the nature and appearance of the information to be displayed. The proposed technique scales down building roofs because they provide much less information than he sides of buildings. The main bodies of buildings are scaled up in compensation. Roof types are simplified to fit into one of four styles. This approach works to maintain detail simplicity, while also retaining the important recognition characteristics of the buildings. Important landmarks are accentuated in the representation as they create visual navigation cues.

(d) Depth Cues: Depth perception is the human ability to segregate 3D visual information into different depth planes. The depth perception of 3D static images is not as natural to depth perception in the real world, so viewers must be provided with 'depth cues', which work to 'trick' vision into interpreting 3D information (MacEachren, 1995). An experiment conducted by Wanger et al. (1992) revealed that shadows provided one of the strongest depth cues (cited in Ware, 2000). Shadows greatly influence the perceived heights of objects (MacEachren, 1995; Ware, 2000; Döllner and Walther, 2003), making them especially suitable for accentuating city building heights. They also promote comprehensibility and naturalism in addition to adding visual interest to the displayed scene (Döllner and Walther, 2003). It is believed that shadows can be correctly interpreted, whether or not they are realistic (Ware, 2000).

The major advantage of this rendering technique is that it possesses familiar attributes. The model resembles a cartoon drawing, which many people are comfortable with. It also features the advantages of a 3D view while not straying too far from the familiarity of a plan view. By having ground features such as roads, paths, railway lines and rivers represented as a plan view, users are able to utilize the model for route planning.

This expressive rendering technique was presented by Döllner at the 2nd Symposium on Location Based Services and TeleCartography in Vienna, 2004. The presentation outlined the potential of these non-photorealistic city models for display on small screen devices (Cartwright, 2004). Considering the severely restricted display area provided on small screens, imagery must be reduced accordingly. Döllner compared a reduced realistic city model with a non-realistic one and found that the latter, whilst considerably smaller than its original size, still managed to remain clearly legible.

5.5.2 Current directions

Whilst all of the above factors are important for the development of this technique, the need to test its effectiveness is paramount. Although theory suggests that this may be an appropriate method of presenting spatial information, it is important to test it for usefulness in real-world situations. The use of mobile devices for navigation is becoming increasingly popular, and it brings with it an interest in the portrayal of 3D spatial information. Current research has focused on realistic representations (Rakkolainen and Vainio, 2001; Vainio et al., 2002; Kray et al., 2003), with little interest in alternative methods of display. If Döllner and Walther's (2003) technique is to become a viable alternative to its photorealistic counterparts, it must be better than them in both technical and utilitarian aspects.

5.6 Assessing the technique

Upon reviewing the state of current state of 3D mobile cartography, and the benefits NPR has brought to computer graphics and other fields a research gap was identified. This led to two main research questions being addressed:

1. Are non-photorealistic computer graphics more effective than photorealistic graphics for the delivery of three-dimensional, spatial information on mobile devices?

2. What is the potential of non-photorealistic computer graphics, combined with three-dimensional cartography, for the representation of spatial information on mobile devices?

In a bid to answer these questions, a prototype utilizing Döllner and Walther's (2003) rendering principles was developed. This prototype was then delivered on a small screen device to simulate its use in this context. By developing a photorealistic prototype using similar methods, an effective comparison could be made between the two techniques. This comparison provided an insight into the strengths and weaknesses of each technique. By employing real users to evaluate the prototype in a real-world setting, valuable user acceptance feedback relating to each technique was gathered. Results achieved through this method were used to identify the most effective rendering technique, whilst also generating recommendations for future development.

5.6.1 Scope of the study

This research aimed at determining whether non-photorealistic computer graphics can be effectively applied to 3D cartography for display on small screen devices. It focused on urban landscapes and a prototype was developed to demonstrate the use of this technique on a specific urban area, and to test the final product on a group of users. It attempted to identify the benefits of non-realistic graphics over realistic graphics for the display of spatial information in this context.

For this research the term '3D' is used extensively. When employed to describe the prototype, 3D refers to planar, two-dimensional figures displayed in 3D axonometric space. In some domains this concept is labelled '2.5D', and for this reason it was deemed important to clarify this relationship. Due the time limitations of this research, a fully working 3D prototype could not be developed.

This research did not propose to develop a NPR technique. Its purpose is to simulate the technique in order to gain a better understanding of the possible benefits of non-photorealistic imagery in cartography. If the findings reveal that NPR techniques could provide significant benefits to 3D cartography, then further research into the most appropriate method would need to be undertaken before the most effective technique can be identified and implemented.

5.6.2 Developing the prototype

The development of the prototype involved three stages: data collection, base map production and preparation for delivery.

5.6.2.1 Data collection

The map coverage of the prototype included the southern part of the city of Melbourne's CBD and the eastern portion of the Southgate Arts and Leisure Precinct. This area was selected because of its popularity as an entertainment destination for both tourists and Melbournians. The area features quite a few of Melbourne's major landmarks, as well as a variety of land cover types, including parkland, varying building densities and forms, different types of pedestrian access routes, and the Yarra River. This provides a reliable basis to ensure that non-realistic 3D maps could be effectively applied to areas with one or all of these physical attributes.

This data collection stage involved gathering the data to construct a base map, as well as information relating to the heights of each building that was to appear in the prototype map. A large-scale vertical aerial photograph covering the area of interest was sourced, and approximate building heights were obtained in the field using a clinometer. Photographs played an important role in the production of the prototype, and these were obtained in the field so that a clear picture of the appearance of the area and each individual building could be established. It was also necessary to locate an appropriate oblique aerial photograph. This image would be used as a realistic comparison to the non-photorealistic map during the evaluation stages.

5.6.2.2 Base map production

The aerial photograph was digitized using *AutoCAD;* this provided a clear base for building footprints. Building heights were extruded according to the clinometer readings. These heights were generalised (i.e. buildings with similar heights in reality were giving the same heights on the map) to keep the map's simplicity as well as its symbolic attributes. This process was basic for uniform building types, where the digitised footprint could easily be extruded to create a solid shape at the appropriate height. More complex buildings needed to be constructed from the top down in order to achieve a recognisable result. Additional details such as windows and roof styles were determined from the photographs taken in the field and applied to the model.

Penguin is a NPR application that runs inside *AutoCAD* to provided seamless rendering without the need for exporting or starting over. *Penguin's* cartoon rendering mode could apply an effective non-photorealistic style to the 3D model created in *AutoCAD,* which could then be output as a 2.5D image at any given viewing angle. Upon experimenting with various line styles and colour schemes a NPR map utilizing the design principles outlined earlier in this chapter was achieved.

An oblique aerial photograph covering the area of interest at a similar angle was sourced and edited so that the final result would provide a realistic comparison for the non-photorealistic map (Fig. 5.5).

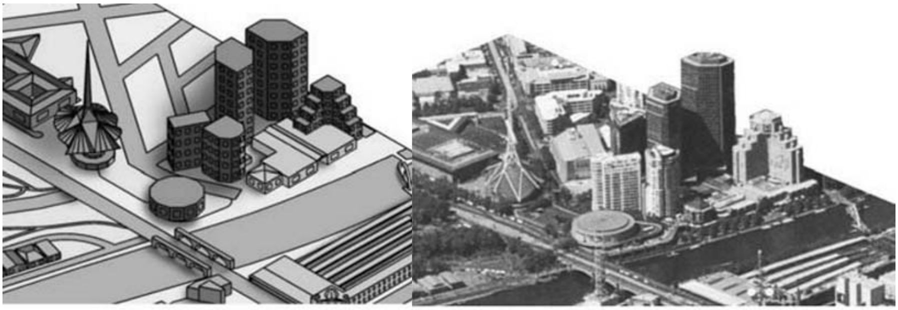


Fig. 5.5. Non-photorealistic and photorealistic maps used for prototype delivery.

5.6.2.3 Preparation for delivery

Flash was used to apply functionality to both the realistic and non-realistic maps in preparation for delivery on a handheld device. Zooming and panning features were incorporated into each map, as well as a labeling function, which could allow users to view the names of major roads and landmarks (Fig. 5.6). Functionality was created identically for each map; the purpose was to have the degree of realism as the only differentiating factor.

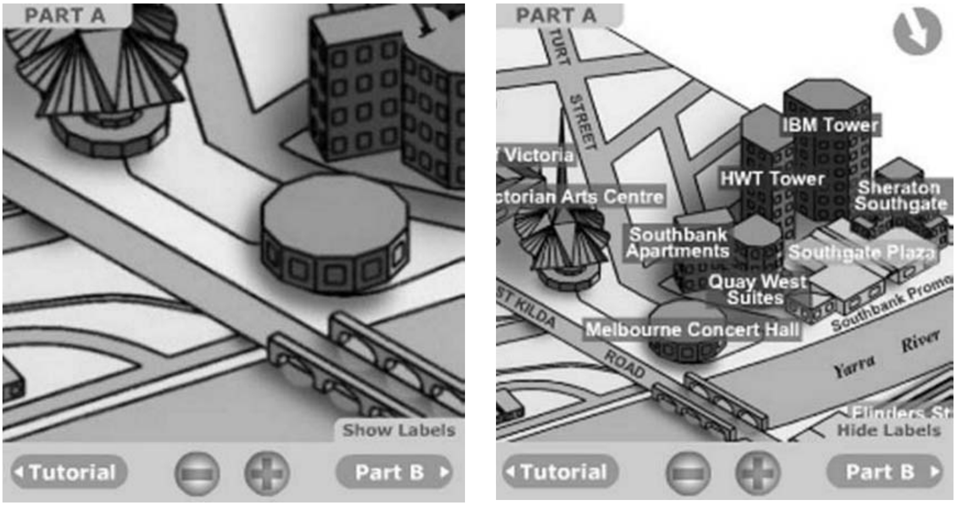


Fig. 5.6. Zoom functionality can be accessed using the + and – buttons. Users are able to pan around the map by clicking and dragging the stylus anywhere within the screen area, and labels can be shown or hidden.

A Compaq iPAQ Pocket PC H3700 PDA running *Microsoft Pocket PC 2002* was used to run the *Flash* enabled maps with a *Flash Player 6 for Pocket PC* plugin. A tutorial map of the world was also created using identical functionality to the other maps. This was intended for users to view prior to the evaluation session to get a feel for interacting with maps on the PDA and *Pocket PC* platform.

5.6.3 User testing and evaluation

Once the prototype was developed it was tested in the area covered by each map. Ten users representative of the target audience of this type of product were invited to participate in this study. Their task was to use the prototype to help them carry out a set of realistic navigational tasks (Table 5.1). All of the users were either second or third year university students and rated their map reading, computer, and handheld device skills as average to good. It was deemed important to include individuals possessing the above criteria as the map design was of utmost interest to this study, rather than the individuals' proficiency with map-related products and interacting with technology. Participants were aged between 18 and 25 years, with 3 females and 7 males volunteering to partake in this study.

The testing procedure employed in this study was not unlike that of a usability study, the only difference was that it was concerned with the usability of the maps themselves rather than the interface. Test materials including an orientation script, background questionnaire and post-test questionnaire were designed to introduce participants to the evaluation and gather data relating to their preferences.

Table 5.1: Navigational tasks undertaken by the test participants.

Task 1 You have just arrived at Finders Street Station and have organised a meeting with a friend who is staying in Southbank. **Use the map to find your way to the front of the Sheraton Southgate Hotel via the Flinders Walk footbridge.**
Task 2 After meeting with your friend, you decide to take a walk along the Yarra River. **Walk up the stairs next to the Sheraton Southgate Hotel and use the map to find your way to Southbank Promenade so you can commence your walk.**
Task 3 You choose to pass the Boat Sheds while on your walk so that you can take some photographs. **Walk along Southbank Promenade and use the map to help you locate the Boat Sheds.**
Task 4 You receive a call from the friend you have recently left. He tells you that he is going to the art gallery and would like to meet you there. **Use the map to find your way from the Boat Sheds to the Art Gallery.**

Each participant was confronted with the test map on the PDA at the commencement of the evaluation. Once they had familiarized themselves with the map's functionality they then went on to proceed to the first task. The session involved four navigational tasks, which presented users with real-world tasks requiring them to identify their location on the map and then traverse a route to a given destination. Each participant had the opportunity to use both the realistic and non-realistic maps – five were required to use the realistic map for the first two tasks and the non-realistic map for the final two tasks and vice versa. This within-subject testing method was employed to ensure that bias did not enter into the results when users transferred the skills they had already obtained in the first two tasks.

After using each map to undertake the predefined tasks, users had enough usage time to develop opinions relating to each map. Upon conclusion of the evaluation session, they were debriefed so that data about their likes and dislikes could be recorded. This data was used to establish a greater understanding of user preferences, which assisted in developing recommendations for 3D map design for small screens.

5.7 Research observations and results

Upon completion of the evaluation session, some constructive observations from the perspective of map development and user preferences could be made.

5.7.1 Map development

The creation of the prototype provided a valuable insight into the different needs and methods required for photorealistic and non-photorealistic city models. Even though the photorealistic map used in the prototype was not created from scratch, some useful observations have been made and can be used to compare these different needs and methods.

The design of the non-realistic map was closely related to that of conventional map design. It utilised the graphic elements of point line and area, and employed visual variables to achieve its symbolic characteristic. While traditional map design is restricted *by* reality, it was found that the design of the realistic map was restricted *to* reality. Features on the map cannot be modified in a way that will compromise their degree of realism. All design elements are already there – what you see in the real world is what will be reproduced.

The non-realistic map developed for the prototype utilised NPR software to produce the final map. The use of such software provides a level of automation in production that is not available to the creation of realistic models. This software also makes modifications hassle-free. A non-expert developer could return to the initial model and easily apply design changes before using the NPR software to output a new design. This also aids in producing map updates, as these can be done quickly and easily. In addition, it is believed that a non-realistic map would require less frequent updates, as real-world changes to building façades and the landscape would not compromise its currency as much as it would a realistic map.

5.7.2 User preferences

It is important that maps are designed to cater for their intended users. The purpose of the map needs to be identified along with the kinds of tasks it will be used for. The information that map users will expect from the product should also be analysed. The map needs to be designed with these considerations in mind in order to achieve maximum user acceptance and understanding.

The test results showed that the non-realistic map was more widely accepted and understood by test participants than the realistic map. The post-test questionnaire (see Appendix A) responses revealed that of the ten participants, eight preferred the non-realistic map. The two participants that preferred the realistic map justified their choice as 'personal preference', and did not offer any negative comments towards the non-realistic map.

Following the field test, users were required to rate their preferences relating to the clarity, usefulness (for the purpose of pedestrian navigation), usability (of each map used during the evaluation session), functionality, aesthetic appeal (according to participants' personal preferences), legibility, appropriateness and innovation of each map. This utilized a Semantic Differential scale, requiring users to indicate an indifferent rating (0) or any rating falling within the negative or positive side of the scale (1-4). The non-realistic map attracted positive ratings across all categories, and was particularly praised for its clarity, usefulness and usability. Ratings for the realistic map were quite neutral across the scale. It achieved a fairly consistent rating across all categories, and did not seem to have any 'stand out' positive attributes when compared to the non-realistic map. There was also a higher incidence of negative ratings across most categories. A comparison of the overall rating values collected from the scaled responses is illustrated in Fig. 5.7.

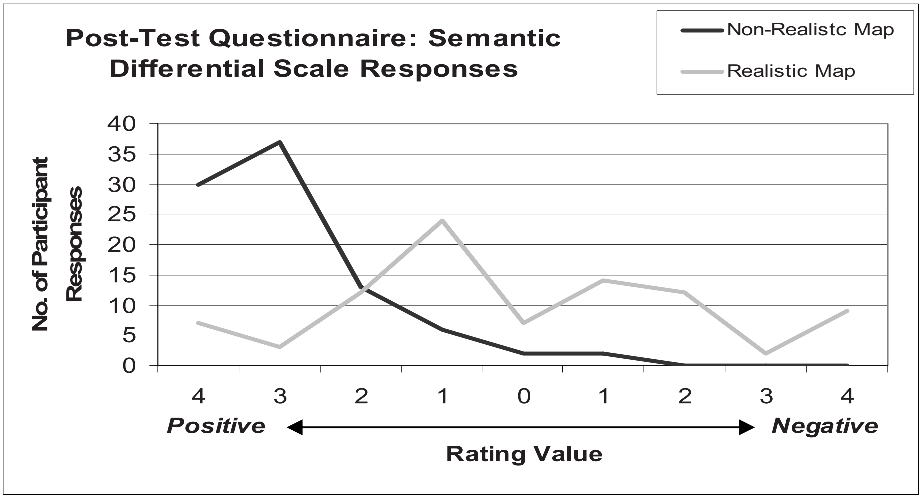


Fig. 5.7. Comparison of positive and negative rating values achieved for each map.

5.8 Research evaluation

This study was ultimately concerned with the design of 3D city maps for delivery on small screen devices. It focused on the creation, delivery and use of non-photorealistic representations in comparison to photorealistic representations, and aimed to determine which method was more effective. To successfully achieve this, the prototype developed for evaluation focused on a specific area of interest and the testing phase utilized a small subset of the product's target users. Even though the prototype's direct focus was of a relatively small-scale nature, the production and testing stages were of great value to this investigation. Whilst the photorealistic map was not created from scratch, background information and observations made during the production of the non-photorealistic map have provided an insight into the processes required for their creation. The area of interest was an inner urban area and featured a wide variety of land cover types, making it safe to imply that similar results would occur if another urban area was used. Likewise, the test participants – being technologically savvy and consumers of other digital forms of mapping applications – closely represented the target users of such a product.

The following conclusions are made based upon the results from this research:

(a) Photorealistic models require extensive detail to achieve the desired result, whereas non-photorealistic models can be created effectively with less detail. Photorealistic models require more time and technical skill during the production phase in order to achieve a high level of realism. Updates are required more frequently for photorealistic models, because even small changes in the real world will compromise their currency. Without compromising quality, the creation of non-photorealistic models is more time and cost efficient when its purpose is to aid pedestrian navigation.

(b) Even though photorealistic city models evaluated were constructed using digital tools, non-photorealistic models are still created using methods that are similar to those employed in conventional map production. The use of conventional processes ensures that map design is familiar to the map designer as well as the map user. This allows them to include the third dimension, without straying too far from 2D production conventions. The design of photorealistic maps is governed by reality, which places enormous restrictions on their design. They are also a relatively new and unauthenticated concept for use for pedestrian navigation.

(c) When applied to producing maps for delivery on handheld devices, 3D non-photorealistic maps are more effective than photorealistic ones. Non-photorealistic maps can be designed to utilise colours that can achieve maximum contrast and legibility for this small-screen delivery medium. On the other hand, delivery may cause a reduction in the quality of a photorealistic map, because slight tonal variations cannot be displayed on mobile devices. If given the chance to use each type of map on a mobile device, it is also likely that users will consider the non-photorealistic map to be more appropriate for small screen display.

(d) In the evaluation results it was noted that users commend non-photorealistic maps for their clarity, usefulness, usability, functionality, aesthetic appeal, legibility, appropriateness, innovation and 'likeability'. Photorealistic maps are particularly weak in the areas of legibility and aesthetic appeal, and achieved unexceptional ratings across all other areas in the evaluation. User preference was indifferent to photorealistic maps, whereas in regard to non-photorealistic maps, users possessed a positive outlook. Non-photorealistic maps were more widely accepted and understood by test users than were photorealistic maps.

(e) Three-Dimensional photorealistic maps provide too much information for pedestrian navigation. It was apparent that photorealistic images provide extra detail not needed by users. Users want to be presented with the most relevant and succinct information, which does not accord with the prime goal of photorealism. Alternatively, non-photorealistic maps presented users with symbolic information, reducing image clutter and enhancing users' ability to extract required information. This provides users with enough information for decision-making in a standardized manner.

5.9 Conclusion

As outlined earlier, most of the studies undertaken make use of photorealistic imagery, and do not attempt to explore alternative methods for the display of 3D city information. More focus needs to be directed at determining the most effective technique to display this information on handheld devices, as current research does not seem to address this issue. It is argued that modern cartography has followed a similar path to that of computer graphics, but as computer graphics realizes the need for NPR in some applications, 3D cartography is still primarily interested in realism. This study was undertaken to determine the potential of non-photorealism for mobile 3D city maps. It is believed that non-realistic 3D maps, when displayed from a bird's-eye view, capture the advantages associated with 3D maps whilst not straying too far from 2D convention.

The research outlined in this chapter aimed at introducing non-photorealistic rendering, a rapidly developing area of interest in the computer graphics community, to 3D mobile cartography. In summary, it can be said, from the results from the study undertaken, that non-photorealistic graphics are more effective than photorealistic graphics for the creation, delivery and use of 3D city maps for pedestrian navigation. This study identified many advantages associated with the use of non-photorealistic imagery and it validated its advantages over photorealism for aiding city navigation using small hand-held devices. Further research is needed that focuses on the most appropriate methods for the development and delivery of non-photorealistic maps. The many benefits of non-photorealism identified in this chapter have provided an initial step for exploring its full cartographic potential.

Appendix A: Post-test questionnaire

Part A

Please indicate your level of agreement with the following statements regarding the non-realistic map:

1. The non-realistic map was clearer than the realistic map.

☐ Strongly Disagree ☐ Disagree ☐ Undecided ☐Agree ☐ Strongly Agree

2. I found that the non-realistic map was better suited to my needs.

☐ Strongly Disagree ☐ Disagree ☐ Undecided ☐Agree ☐ Strongly Agree

3. The non-realistic map was designed appropriately for viewing on a small screen.

☐ Strongly Disagree ☐ Disagree ☐ Undecided ☐Agree ☐ Strongly Agree

4. I did not favour the non-realistic map.
☐ Strongly Disagree ☐ Disagree ☐ Undecided ☐Agree ☐ Strongly Agree

5. The non-realistic map was more aesthetically pleasing.
☐ Strongly Disagree ☐ Disagree ☐ Undecided ☐Agree ☐ Strongly Agree

Using the following rating scales, please circle the number nearest the term that most closely matches your feeling about the non-realistic map:

Unclear . . . . 3 . . . 2 . . . 1 . . . 0 . . . 1 . . . 2 . . . 3 . . . Clear

Useless . . . . 3 . . . 2 . . . 1 . . . 0 . . . 1 . . . 2 . . . 3 . . . Useful

Unusable . . . . . . 3 . . . 2 . . . 1 . . . 0 . . . 1 . . . 2 . . . 3 . . . Usable

Non-Functional . . 3 . . . 2 . . . 1 . . . 0 . . . 1 . . . 2 . . . 3 . . . Functional

Appealing. . . . . . 3 . . . 2 . . . 1 . . . 0 . . . 1 . . . 2 . . . 3 . . . Unappealing

Illegible . . . . 3 . . . 2 . . . 1 . . . 0 . . . 1 . . . 2 . . . 3 . . . Legible

Inappropriate . . . 3 . . . 2 . . . 1 . . . 0 . . . 1 . . . 2 . . . 3 . . . Appropriate

Conventional. . . . 3 . . . 2 . . . 1 . . . 0 . . . 1 . . . 2 . . . 3 . . . Innovative

I like . . . . 3 . . . 2 . . . 1 . . . 0 . . . 1 . . . 2 . . . 3 . . . I dislike

Part B
Please indicate your level of agreement with the following statements regarding the realistic map:

1. The realistic map was easier to use.
☐ Strongly Disagree ☐ Disagree ☐ Undecided ☐Agree ☐ Strongly Agree

2. The realistic map appeared to be cluttered.
☐ Strongly Disagree ☐ Disagree ☐ Undecided ☐Agree ☐ Strongly Agree

3. I preferred the realistic map.
☐ Strongly Disagree ☐ Disagree ☐ Undecided ☐Agree ☐ Strongly Agree

4. The realistic map did not provide the most useful information.
☐ Strongly Disagree ☐ Disagree ☐ Undecided ☐Agree ☐ Strongly Agree

5. I did not think the realistic map was suitable for viewing on a small screen.
☐ Strongly Disagree ☐ Disagree ☐ Undecided ☐Agree ☐ Strongly Agree

Using the following rating scales, please circle the number nearest the term that most closely matches your feeling about the realistic map:

Unclear . . . . 3 . . . 2 . . . 1 . . . 0 . . . 1 . . . 2 . . . 3 . . . Clear

Useless . . . . 3 . . . 2 . . . 1 . . . 0 . . . 1 . . . 2 . . . 3 . . . Useful

Unusable . . . . . . 3 . . . 2 . . . 1 . . . 0 . . . 1 . . . 2 . . . 3 . . . Usable

Non-Functional . . 3 . . . 2 . . . 1 . . . 0 . . . 1 . . . 2 . . . 3 . . . Functional

Appealing. . . . . . 3 . . . 2 . . . 1 . . . 0 . . . 1 . . . 2 . . . 3 . . . Unappealing

Illegible . . . . 3 . . . 2 . . . 1 . . . 0 . . . 1 . . . 2 . . . 3 . . . Legible

Inappropriate . . . 3 . . . 2 . . . 1 . . . 0 . . . 1 . . . 2 . . . 3 . . . Appropriate

Conventional. . . . 3 . . . 2 . . . 1 . . . 0 . . . 1 . . . 2 . . . 3 . . . Innovative

I like . . . . 3 . . . 2 . . . 1 . . . 0 . . . 1 . . . 2 . . . 3 . . . I dislike

Part C

1. Please indicate your personal preference by circling the corresponding number on the scale below:

Realistic Map ←——————————————→ Non-Realistic Map

5 4 3 2 1 Neutral 1 2 3 4 5

2. Which map did you find the most useful?
☐ Realistic ☐ Non-Realistic ☐ Undecided

3. Please state the reasons why you preferred the map you selected in Q12:

____

____

____

4. Please state the reasons why you did not find the other map as useful:

____

____

____

5. Do you have any other comments or suggestions?

____

____

____

References

Bieber, G. and Giersich, M. (2001): 'Personal mobile navigation systems - Design considerations and experiences', *Computers and Graphics (Pergamon),* **25**(4), pp. 563-570.

Black, J. (2003): *Visions of the World.* Mitchell Beazley, London.

Cartwright, W. 2004, *Personal communication.*

Christie, J., Klein, R. M. and Watters, C. (2004): 'A comparison of simple hierarchy and grid metaphors for option layouts on small-size screens', *International Journal of Human-Computer Studies,* 60(5-6), pp. 564-584.

Collinson, A. (1997): 'Virtual worlds', *Cartographic Journal,* 34(2), pp. 117-124.

Döllner, J. and Walther, M. (2003): 'Real-time expressive rendering of city models'. *Proceedings Seventh International Conference on Information Visualization (IV '03).* London, England, July 16 - 18. IEEE Computer Society, pp. 245-250.

Duke, D. J., Barnard, P. J., Halper, N. and Mellin, M. (2003): 'Rendering and Affect', *Computer Graphics Forum,* 22(3), pp. 359-368.

Durand, F. (2002): 'An Invitation to Discuss Computer Depiction', *Symposium on Non-Photorealistic Animation and Rendering (NPAR 2002).* Annecy, France, 3-5 June, pp.111-124.

Elliot, J. (1987): The City In Maps: Urban Mapping to 1900. The British Library, London.

Feiner, S., MacKinlay, J., Blinn, J., Greenberg, D. and Hagen, M. (1988): 'Designing effective pictures: Is photographic realism the only answer?' (panel). *ACM SIGGRAPH 1988,* 22(4), p.351.

Ferwerda, J. A. (2003): 'Three varieties of realism in computer graphics'. *Proceedings of the SPIE - The International Society for Optical Engineering.* Santa Clara, CA, USA, Jan 21-24. pp. 290-297.

Finkelstein, A. and Markosian, L. (2003): 'Nonphotorealistic rendering', *IEEE Computer Graphics and Applications,* **23**(4), pp. 26-27.

Gershon, N. D., Braham, R., Fracchia, D., Glassner, A., Mones-Hattal, B. and Rose, R. (1996): 'Breaking the myth: One picture is NOT (always) worth a thousand words'. *Proceedings of the ACM SIGGRAPH Conference on Computer Graphics.* New Orleans, LA, USA, 4-9 August, pp. 491-492.

Goldstein, D. (1999): 'Intentional non-photorealistic rendering', *Computer Graphics,* **33**(1), pp. 62-63.

Gooch, A. A. and Willemsen, P. (2002): 'Evaluating Space Perception in NPR Immersive Environments', *2nd International Symposium on Non-Photorealistic Animation and Rendering (NPAR '02).* Annecy, France, 3-5 June, pp. 105-110.

Gooch, B. and Gooch, A. (2001): Non-Photorealistic Rendering. A K Peters, Natick, Mass.

Graham, C., Urquhart, K., Davies, J. and Vetere, F. (2003): 'Evaluating and evolving digital mobile maps'. *OzCHI2003: New directions in interaction, information environments, media and technology.* Brisbane, Australia, 26-28 November. Information Environments Program, pp. 116-125.

Gregory, R. L. (1998): Eye and Brain: The Psychology of Seeing. 5th ed., Oxford University Press, Oxford.

Häberling, C. and Hurni, L. (2002): 'Mountain cartography: revival of a classic domain', *ISPRS Journal of Photogrammetry & Remote Sensing,* **57**, pp. 134-158.

Hagen, M. 1986, *Varieties of Realism.* Cambridge University Press, Cambridge.

Halper, N., Schlechtweg, S. and Strothotte, T. (2002): 'Creating Non-Photorealistic Images the Designer's Way', *NPAR Symposium on Non-Photorealistic Animation and Rendering (NPAR '02).* Annecy, France, 3-5 June*, pp.* 97-104.

Herman, I. and Duke, D. (2001): 'Minimal graphics', *IEEE Computer Graphics and Applications,* 21(6), pp. 18-21.

Hodgkiss, A. G. (1981): Understanding Maps. Dawson, Kent.

Jones, C. B. (1997): Geographical Information Systems and Computer Cartography. Longman, Harlow, England.

Keates, J. S. (1989): Cartographic Design and Production. 2nd ed., Longman Scientific & Technical, Essex.

Konig, H., Schneider, J. and Strothotte, T. (2000): 'Haptic exploration of virtual buildings using non-realistic haptic rendering'. *7th International Conference on Computers Helping People with Special Needs (ICCHP).* Karlsruhe, Germany, 17-21 July, pp. 377-384.

Kray, C., Laakso, K., Elting, C. and Coors, V. (2003): 'Presenting route instructions on mobile devices', *International Conference on Intelligent User Interfaces, Proceedings (IUI),* Miami, FL, USA, 12-15 January, pp. 117-124.

Lum, E. B. and Ma, K.-L. (2002): 'Interactivity is the key to expressive visualization', *ACM SIGGRAPH Computer Graphics,* 36(3), pp. 5-9.

MacEachren, A. M. 1995, *How Maps Work.* New York.

Markosian, L., Kowalski, M. A., Trychin, S. J., Bourdev, L. D., Goldstein, D. and Hughes, J. F. (1997): 'Real-time nonphotorealistic rendering', *Proceedings of the ACM SIGGRAPH Conference on Computer Graphics.* Los Angeles, CA, USA, 3-8 August, pp. 415-420.

Mignotte, M. (2003): 'Unsupervised statistical sketching for nonphotorealistic rendering models', *IEEE International Conference on Image Processing.* Barcelona, Spain, September, pp. 573-576.

Monmonier, M. S. (1982): *Computer-assisted cartography : principles and prospects.* Prentice-Hall, New Jersey.

Muehrcke, P. C., Muehrcke, J. O. and Kimerling, A. J. (2001): *Map Use.* JP Publications, Madison.

National Science Foundation (2004): 'Visualization: a way to see the unseen'. http://www.nsf.gov/od/lpa/news/publicat/nsf0050/visualization/visualization.htm. Web page accessed: 18th June 2004

Patterson, T. (1999): 'Designing 3D landscapes'. *In:* Peterson, M. P., Cartwright, W. and Gartner, G. F. (Eds.) *Multimedia Cartography*. Springer, New York, pp. 217-229.

Raisz, E. 1948, *General Cartography.* McGraw-Hill, New York.

Rakkolainen, I. and Vainio, T. (2001): 'A 3D city info for mobile users', *Computers and Graphics (Pergamon),* 25(4), pp. 619-625.

Robinson, A. H. c1995, *Elements of Cartography.* Wiley, New York.

Schumann, J., Strothotte, T., Raab, A. and Laser, S. (1996): 'Assessing the effect of nonphotorealistic rendered images in CAD'. In R. Bilger, S. Guest, and M.J. Tauber (Eds.), *Proc. Computer Human Interaction (CHI'96),* ACM Press, pp. 35-42.

Sevo, D. (2000): 'History of Computer Graphics'. http://hem.passagen.se/des/hocg_intro.htm. Web page accessed: 18th June 2004.

Swanson, J. (1999): 'The cartographic possibilities of VRML'. *In:* Peterson, M. P., Cartwright, W. and Gartner, G. F. (Eds.) *Multimedia Cartography*. Springer, New York, pp. 181-194.

Vainio, T., Kotala, O., Rakkolainen, I. and Kupila, H. (2002): 'Towards scalable user interfaces in 3D city information systems', *Human Computer Interaction with Mobile Devices. 4th International Symposium, (Mobile HCI 2002). Proceedings (Lecture Notes in Computer Science Vol.2411),* pp. 354-358.

Ware, C. (2000): Information Visualization. Morgan Kaufmann, San Francisco.

Wildbur, P. (1989): Information Graphics. Van Nostrand Reinhold Company, New York.

6 Context-Aware Applications Enhanced with Commonsense Spatial Reasoning

Matteo PALMONARI , Stefania BANDINI

Department of Computer Science, Systems and Communication (DISCo), University of Milan - Bicocca

Abstract. A major issue in Pervasive Computing in order to design and implement context–aware applications is to correlate heterogeneous information acquired by distributed devices to provide a more comprehensive view of the context they inhabit. Although information can be geo-referenced according to quantitative models there are a number of reasons for which Qualitative Spatial Representations can be preferred in such context. The paper presents a knowledge-based approach to correlation of information coming from different sources based on Logical Commonsense Spatial Reasoning. In particular a class of models that can be exploited for reasoning about correlation is presented and a framework to provide the desired inferences within a Hybrid Logic framework is given. This framework is claimed to be enough flexible to be exploited in different application domains and an example for a Smart Home application is discussed.

6.1 Introduction

Thanks to the improvement and growing availability of information acquisition and delivery technology (sensors, personal devices, wi-fi, and so on), the computational power can be embedded almost in every object populating the environment. This brought a growing attention on pervasive and ubiquitous systems. These systems are characterized by different - possibly mobile - components distributed in the environment; they are basically devoted to collect, process and manage information in order to support users in different kind of activities (ranging from monitoring and control of specific areas to management of personal data, and so on) (Zambonelli and Parunak, 2002). Applications aiming at being proactive and at reducing users' intervention need to be aware of the context in order to both adapt their behavior and meet users' expectations delivering specific contents and taking proper actions (Dey, 2001).

A first concern for these systems is related to the possibility of ubiquitous access and provision of information, and the research area focusing on this aspect is generally referred to as Ubiquitous Computing. A second concern is related to the opportunities provided by new information acquisition technologies of acquiring and processing information more and more pervasively. When the major focus is on this last issue, which concerns a massive exploitation of sensors and ambient intelligent technology, the research area addressed is generally referred to as Pervasive

Computing. Context awareness is related to both Pervasive and Ubiquitous Computing: if the context in which applications operate dynamically changes since information is ubiquitously accessed, contextual information can be acquired mostly thanks to information acquisition technologies.

Contextual information concerns users, e.g. users' preferences, but also the technological and physical environment (e.g. the presence of other devices and their properties, the availability of services, the spatial environment in which the application takes place, and so on). Perceiving, representing and manipulating contextual information is necessary to perform high-level tasks that devices need to carry out in order to behave as much autonomously as possible, according to the basic goal of Pervasive Computing.

Therefore, a major issue for the design and the implementation of context–aware pervasive applications concerns the correlation of heterogeneous information acquired by distributed devices in order to provide a more comprehensive view of the context they inhabit. Here, extending the work presented in Bandini et al. (2005b), we present a knowledge-based approach to correlation of heterogeneous information coming from different sources based on Knowledge Representation techniques for qualitative spatial reasoning.

In particular, within a conceptual architecture discussed in Bandini et al. (2004), we define a general strategy to correlation of *events* in Pervasive Computing domains. The strategy consists of three main steps: the choice of a spatial model to represent the application environment, the choice of a spatial logic to reason on the defined model, and the definition of correlation axioms to establish logical and spatial correlation among events in order to infer the interesting *scenarios*.

The chapter is organized as follows. The knowledge-based approach is presented in the next section; the section introduces a conceptual architecture for information processing in Pervasive Computing and proposes a spatial representation-based strategy for the correlation of information coming from different sources. After discussing why Qualitative Spatial Representation and Reasoning (QSRR) is attractive for these application contexts and the main approaches developed by the QSRR community, section 6.3 introduces a class of qualitative spatial models, namely Commonsense Spatial Models, whose primitives are the notions of *place* and *commonsense spatial relation*. On the basis of the formal properties that characterize classes of spatial relations (proximity, containment and orientation), the more specific class of Standard Commonsense Spatial Models is defined. Section 6.4 presents spatial hybrid logics as a powerful and flexible framework for Commonsense Spatial Reasoning, and, in particular to reason about correlation on top of Commonsense Spatial Models. Finally, section 6.5 discusses an example in which the general strategy, the models and the logical framework introduced are applied to reason about correlation of alarms in a Smart Home domain. Concluding remarks end the chapter.

6.2 Knowledge-based correlation of information with spatial representation and reasoning

In Pervasive Computing, information on the environment provided by acquisition devices may loose significance as a huge number of sensors tend to produce an overload of information. On the one hand, this problem is related to those systems, e.g. Control and Monitoring Systems (Bandini et al. 2004), which are explicitly devoted to support the interpretation of collected data. On the other hand, this correlation is also necessary to develop context aware applications endowed with enough "intelligence" to go beyond the notification of plain information acquired by sensors. In fact, also for providing ambient intelligence or setting up a Smart Environment, data dynamically acquired by sensors must be integrated in order to select and define proper actions supporting users in a more proactive way. Information produced by sensors or collected via communication are a relevant part of context of which application are supposed to be aware of. From this perspective, with respect to correlation, experience with Monitoring and Control Systems can be paradigmatic.

6.2.1 A knowledge-based approach

The integration of information coming from distributed sources is often intended as information fusion; this integration is usually tackled by means of non knowledge-based techniques, resulting often more efficient for specific purposes than knowledge-based ones (Carvalho et al., 2003). Nevertheless, the solutions adopted by means of powerful techniques such as stochastic-based ones are often domain dependent and calibrated on the application at hand. In this sense, in order to gain in generality and to provide a framework to capture the main traits involved in the correlation tasks, it could be worth inquiring a knowledge-based approach; such an approach, eliciting the underlying representational model, forces to focus on the knowledge model applied for correlating information. For these reasons a knowledge-based approach may be particularly suitable when the phenomena to be discovered are known, when there is some knowledge about *how* the correlation must be carried out, and this knowledge is difficult to be extracted from a set of raw data.

As far as Control and Monitoring Systems are concerned, knowledge-based approaches have been successfully applied also in very critical domains such as in traffic management (e.g. see Bandini et al., 2002; Ossowski et al., 2004); in such contexts, knowledge about the interesting correlations are often provided by domain experts and hence coded into a formal knowledge representation system in order to support reasoning. Nevertheless, an increasing attention on semantic and well structured representations of context (including a representation of the environment) has strongly characterized recent research on context awareness (e.g. see the ontology-based approaches of Chen et al. (2004) and Christopoulou et al. (2005)); semantics is in fact supposed to favour context awareness enabling interoperability, and supporting the definition of high-level criteria for information management, eventually customizable and specifiable by users (Chen et al., 2004).

Our approach to information correlation follows the perspective of those working on high-level semantic context models with Knowledge Representation techniques (e.g. with ontologies) in order to provide an integration layer on the top of other processing techniques. The integration between numeric-intensive techniques for data interpretation and knowledge-based models can be supported by a conceptual framework presented in Bandini et al. (2004) for Monitoring and Control Systems.

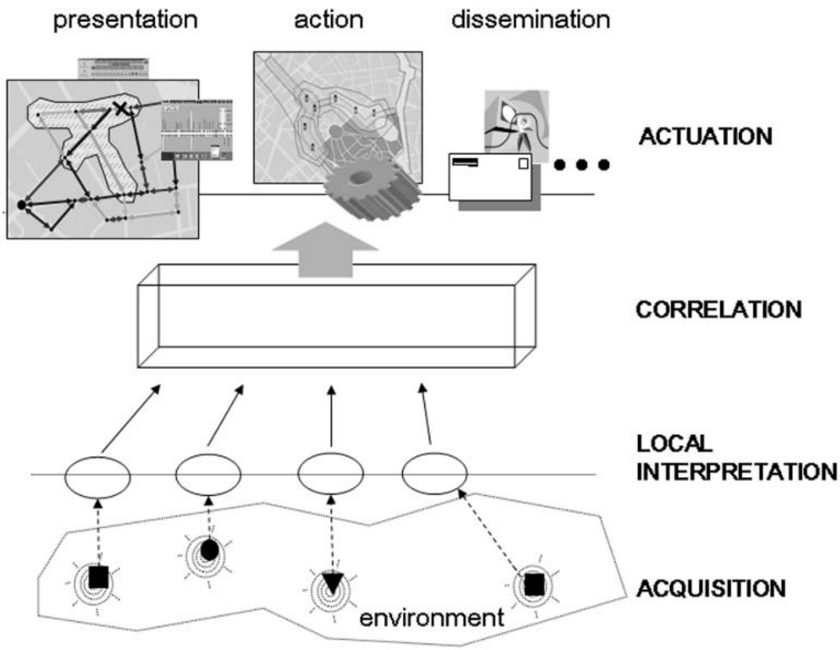


Fig. 6.1. The four level architecture

The framework, which is straightforwardly generalizable to pervasive systems devoted to collect and interpret data, introduces a conceptual architecture that is sketched in Fig. 6.1 and consists of four levels:

1. the **acquisition** level - sensors and devices, eventually different and heterogeneous, acquire data from the environment or from other devices (e.g. a sensor analyse air composition to detect the presence of smoke);

2. the local **interpretation** level - data acquired by sensors are processed and interpreted with respect to their local models[3], returning information about a specific parameter or about a particular portions of the environment (e.g. a piece of information as "smoke" is associated to the sensor activity, the detection of the presence of a person is the result of image processing algorithms on the data acquired by sensors);
3. the **correlation** level - information coming from local interpretations, and possibly from different sources, is correlated, that is it is managed and filtered according to a more global view of the whole situation (e.g. neither a broken sensor detection nor the presence of a person trigger an alarm to the surveillance center, but the joint combination of both the alarms is interpreted as the evidence for a dangerous situation);
4. the **actuation** level - different actions are taken on the basis of the available information (e.g. an alarm is sent to the surveillance center, a traffic regulation plan is activated, a thematic map presenting high-level information about the monitored area is displayed).

A concrete example of this integration between knowledge-based and intensive algorithmic techniques is given by SAMOT, a system devoted to traffic monitoring over a highway; in this application pictures acquired by video-cameras are processed by genetic algorithms and the correlation layer has been implemented with a production rule system (Bandini et al., 2005a).

The result of the first local processing consists in a piece of information which is minimally significant, and which, therefore, can be encoded as a report of what sensors detected. This information, which go beyond raw data acquired by sensors, can be homogeneously represented as a set of *events*. In a Smart Home example, local interpretations may report events such as "smoke in the kitchen", "person detected near the entrance", "temperature is 30°C", and so on. *Events* have a *location* and possibly *duration*; in Pervasive Computing domains as far as events are a result of local processing over data acquired by sensors, the duration is often replaced by a time stamp relating an event to its detection time. Space and time are therefore those aspects of information on the basis of which heterogeneous information can be considered and correlated.

In this chapter we focus on spatial correlation, and successively we discuss how the approach can be extended to consider also the temporal dimension, and the problems arising from a Knowledge Representation point of view when time is considered. This is reasonable with respect to context awareness since one can assume to consider what is known at a given moment, referring to a set of events occurring at that time (time can be handled implicitly, outside the inference system).

A knowledge-based approach allows to consider arbitrary events coming from heterogeneous sources of information and to exploit a homogeneous representation

[3] Very often, local interpretation are performed locally by the acquisition devices themselves which can be equipped with suitable software (e.g. a camera endowed with a video image processing software); nevertheless, we still consider "local" an interpretation that is based only on local data and parameters, also when processing is performed elsewhere (e.g. if a video image processing software runs in a control center but analyses images taken by a single camera).

to model correlation as logical inference. An advantage coming from the representation of arbitrary events as spatially (and temporally) referenced information is that the introduction of new kinds of events (e.g. environmental pollution information generated by a new subsystem of sensors) would not require a re-definition of the whole model, but only the definition of a new set of correlation formulas.

6.2.2 Correlation with spatial reasoning

Our knowledge-based approach to correlation of spatially referenced information can be therefore characterized as *correlation of events* produced by local interpretations according to domain specific principles. Information provided by sensors or referred to specific subparts of the environment must be related to a global view that goes beyond the local view proper of the immediately available information. This "more global view" can be captured by the notion of *scenario*, where a scenario can be defined as the specification of significant logical correlations among local descriptions.

In particular, the spatial dimension must be taken into account, establishing connections among pieces of information according to the spatial relationships holding among their spatial references. Since the identification of significant scenarios strongly relies on a spatial model, correlation can be carried out as a form of spatial reasoning, where the spatial inferences support the identification of significant *spatial scenarios*.

But, in order to support reasoning a flexible formal language for talking about events referred to different classes of spatial models is needed. When correlations are defined as formulas of a logical language, inferences about events and scenarios can be supported by the inferential mechanisms of the logic itself.

A general *tree-step strategy* can be therefore defined as follows:

- choose a suitable spatial model to represent the spatial environment of an application according to the demand of the correlation task;
- choose a suitable logical language expressive enough to talk about spatially referenced information (that can be considered as a set of events), and about spatial relations among the entities involved;
- define the formulas representing logical correlations among events in the logical spatial language adopted in order to exploit the spatial logic's deductive power to infer scenarios from local and primitive information.

In the following we discuss the kinds of spatial models that could be used in this context, and in particular explaining why qualitative spatial models are recalled. Therefore we provide an approach to spatial modeling defining a class of qualitative spatial models and a modal-like logic to reason over them, which offers particular flexibility and versatility in order to be applied to different application contexts.

6.3 Commonsense spatial models for information correlation

According to the general strategy based on space presented above, a major problem consists in choosing a good spatial representation and reasoning framework in order to define both a class of spatial models and a logic (a language, its semantics and its calculus) enabling reasoning about information correlation based on such a class of models. Of course there is a variety of spatial models used for different purposes in computer science. As for Pervasive Computing, GPSs are based on a coordinate based geodetic model and current GISs use vector based and raster models. These models can be exploited to represent spatial information (e.g. as position), to query spatial databases and to carry out different mathematical computation.

Nevertheless, when it comes to “reasoning”, that is, to perform some kinds of inference about spatial entities and with spatial concepts, there are at least three good reasons to take into account qualitative spatial models. First, reasoning with quantitative models is often intractable (Bennett, 1996). Second, quantitative models such as mathematics of Euclidean space are often too precise for the purpose at hand (Cohn and Hazarika, 2001). This second issue is twofold on its turn: from the one hand a full Euclidean representation may be over-sized with respect to the kind of inference expected, with repercussions on the computational aspects of the representation and reasoning system; on the other side, information about space is not always enough precise to be mapped into a full geometrical model (Bennett, 1996). This means that it is not always possible to build a full and precise geometrical representation on the basis of the available information (e.g. position reference can be vague).

The third issue is particularly relevant with respect to correlation: the spatial reference relevant to correlation is often related to a semantic spatial model, rather than to a geometrical and mathematical one. As an example, consider a smart application enabling some Personal Digital Assistants (PDA) to adapt the users’ profiles on the basis of locations selected by users; in order to activate a profile on a PDA, the fact that the user is located into the University (i.e. the place where a number of services are available) seems much more relevant than the position specified in terms of coordinates, since humans often relate to space in a qualitative way. This means that, in order to enable interaction between the users and the spatial representation exploited for referencing and correlating information, there is the need to bridge the gap between the users’ representations of space and the application’s spatial data structure.

6.3.1 Qualitative spatial representation and reasoning: related work

Research on Qualitative Spatial Representation and Reasoning (QSRR) has been intense in the last years and has been carried out in different application domains; in particular, with respect to Pervasive Computing, different researches addressed qualitative spatial inferences within GIS and spatial databases. Accurate overviews of the different approaches and techniques proposed in the QSRR area are given by Cohn and Hazarika (2001), by Egenhofer and Mark (1995), and Fonseca et al. (2002); the latter two works focus on qualitative reasoning in GIS.

Most influent QSRR approaches are based on mathematical topology. Different qualitative topological and mere-topological (Casati and Varzi, 1999) theories have been discussed and formalized. The Region Connection Calculus (RCC) is probably the most influent theory developed in the field of qualitative spatial representation and reasoning. RCC has been formalized in First Order Logic (Randell et al., 1992), but, since such an axiomatization generates undecidable results, tractable sets of inferences have been identified and have been formalized into decidable *modal logics*, e.g. by Bennett (1996) and Aiello and van Benthem (2002).

With respect to spatial reasoning for the recognition of significant scenarios, these approaches seem over-sized, and in fact they have been applied to problems different from correlation. Topological theories such as the RCC represent space as a set of spatial regions that can be *connected* in different ways on the basis of their internal structure, focusing on inferences of topological relations holding between regions; this is made possible by the consideration of the internal structure of spatial entities that can be generally described in terms of still qualitative concepts such as "interior", "complement" and "boundary". According to Cohn and Hazarika (2001), most of reasoning in this field is based on the so called *transitivity tables*: such tables allow to infer, given a relation holding between two regions A and B, and a relation holding between B and C, the set of possible relations holding between A and C. On the other hand, in the above mentioned spatial modal logics, formulas of the language are interpreted as spatial regions within a topological framework.

With respect to the aim of correlating information identifying significant scenarios, neither the derivation of spatial relations from the internal structure of topological regions (part of this knowledge may be known by designed or obtained by technological tools) nor the execution of inferences on the basis of a transition tables is relevant. Of course, some inferences based on the meaning of the spatial relations involved need to be granted, e.g. exploiting the transitivity of the relation "to be in", but such inferences need not to be necessarily justified, in the last resort, on topological properties of the involved entities. Consider for example that a piece of information about a device which is located in a park can be provided by a localization system rather than inferred from the spatial representation of the two entities (the device and the park); again, two rooms can be considered connected when it is possible to access one from the other one and not on the basis of a shared boundary.

Starting from these considerations, our approach focuses more on the formal characterization of the meaning of spatial relations (this characterization is what makes it possible to perform inference), than on the foundation of such a meaning with respect to a strong ontology of space.

A simple idea to develop a qualitative spatial model of space supporting reasoning about the environment consists in identifying a set of places, i.e. a set of interesting spatial entities, and a set of different spatial relations holding among them. These models are referred to as Commonsense Spatial Models (CSM). Basically, a CSM can be considered as a spatial relational structure.

Consider that, given a spatial relation, its extensional meaning is defined, according to the standard logical approach, as the set of tuples for which the relation holds. Obviously, this provides little help to support inference: the possibility to perform inference is related to the *intensional* meaning of the relations. The intensional meaning is given by high-level properties of relations formally specified within the language. For example, some inferences about the relation “in” are made possible when a formal property such as *transitivity* is defined by a logical axiom and can be used by an inference engine.

Therefore, discarding any foundational perspective over spatial reasoning, the key point in developing our spatial representation approach to correlation of information consists in working on this relationship among spatial models and the logic to reason over them, in order to support the joint definition of both spatial representations and a logical language to reason about them. A logic that provides this framework is Hybrid Logic, an extension of Modal Logic that significantly empowers its reasoning capabilities.

According to the above considerations, various classes of inferences can be supported assuming *places* as the primitive spatial entities (that is, notions not further analysed), provided that the meaning of the spatial relations involved is formally specified. Rather, topological connection relations are not sufficient and it is more important to provide a compact model combining a number of different kinds of spatial relations. Observe that the relational structures proposed are close to those graph-like models widely used in Pervasive and Ubiquitous Computing from less theoretically oriented approaches. Moreover, if one prefers a more grounded topological approach, then Hybrid Logic is enough expressive to define different topological theories.

The strategy consists in starting with the general definition of what we consider a Commonsense Spatial Model, successively analysing a specific class of Commonsense Spatial Models according to the formal properties of particularly significant spatial relations. Therefore, we discuss how the class of models introduced can be easily represented within Hybrid Logic, showing that a logical calculus can be defined to reason about them; the formal details involved in this passage can be found in Bandini et al. (2005c) and therefore are omitted. Instead, we show how the flexibility provided by Hybrid Logic allows to easily extend the logic to consider Spatial Models with different other relations.

6.3.2 Commonsense spatial models

Let us start with a simple example. Suppose to have a sensor platform installed in a building in order to monitor a significant portion of it (and, eventually, to take suitable control actions); an example of apartment populated by a number of sensors is represented in the two left-most images of Fig. 6.2. Sensors distributed in the environment return values that can be interpreted in order to provide *local* descriptions of what is happening in the range of each sensor, possibly generating alerts or alarms. According to the four-level architecture introduced in section 6.2.1, these local descriptions can be interpreted as events with a spatial reference.

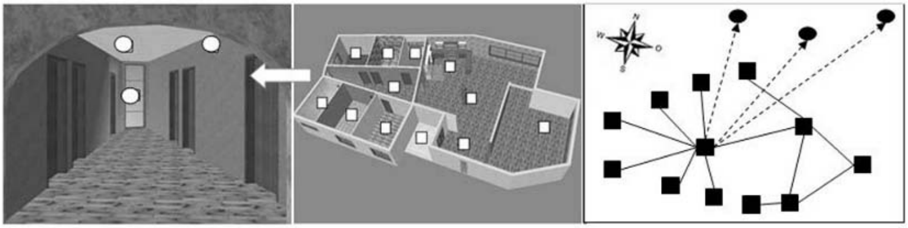


Fig.6.2. A commonsense spatial model of a monitored apartment with sensors distributed in the rooms.

A Commonsense Spatial Model supporting reasoning about the environment can be defined as a finite topology whose nodes are interesting places and whose relations are commonsense spatial relations (CSR) arising from an abstraction of the spatial disposition of these places. From a formal point of view, CSMs can be defined as relational structures according to the following definition:

Definition 1. *A Commonsense Spatial Model* $CSM = \langle P, R_S \rangle$ *is a relational structure, where* $P = \{p_1, ..., p_k\}$ *is a finite set of places, and* $R_S = \{r_1, ..., r_n\}$ *is a finite non-empty set of binary conceptual spatial relations labeled by a set of labels* $L \subset$ □, *and where, for each* $i \in L$, $R_i \subseteq P \times P$.

A place here is taken as a conceptual entity completely identified by the aggregation of its properties, which may concern the type of place (e.g. a place can be a sensor or a room), the internal status of the place (e.g. "is_faulty", "is_working"), its functional role (e.g. a kitchen or a living room), and so on. From a conceptual point of view, *R* can be any arbitrary set of binary CSRs; nevertheless, some classes of relations significant for wide classes of reasoning domains are analytically and formally characterized in the following paragraphs. This lack of constraints on what can be taken as a CSR, may be seen as a weakness of the model, but is related to the main principles guiding this approach. In fact, in comparison with other well known topological models (e.g. the RCC calculus of Randell et al. (1992), and the spatial modal logic of Aiello and van Benthem (2002)), it is neither possible nor useful to identify a minimal set of primitive relations. Indeed, this approach is not aimed at providing a mathematical model of space, but at defining the basic elements for the specification of axioms characterizing relevant properties of specific environments.

6.3.3 Classes of commonsense spatial relations and standard CSM

As already mentioned, there are some significant classes of relations that play a special role in the definition of a commonsense model of space. On the basis of both theoretical and pragmatic observations, three main classes of relations can be identified according to a set of shared formal properties: the classes of *proximity*, *containment* and *orientation* relations.

Proximity. A first basic class of relations establishes connections among spatial entities. A proximity relation RP is a reflexive and symmetric relation holding between two places *p* and *q* when the place *q* is directly reachable from place *p* (without passing through another place). Different criteria of reachability, both physical and metaphorical, can be adopted to define a proximity relation (another example of proximity relation could be networking among devices in distributed systems).

Containment. Location is a key spatial concept, especially in Pervasive Computing domains. Moreover, in our approach, places represent arbitrary entities possibly having different shapes, dimensions and nature (e.g. a room and a printer are both places); an inclusion relation over places is therefore needed also in order to relate different types of places: an object may be *inside* a room that may be *inside* a building, and so on. Since our places have zero dimension and we are basically interested in the inference power enabled by our characterization, location and inclusion can be treated in a homogeneous way, identifying a unique class of containment relations. Defining containment relations as typical mereological relations (they are *reflexive*, *antisymmetric* and *transitive* relations establishing partial orders), a relation $R_{IN}(p,q)$ can be interpreted as stating that the place *q* is *contained* in the place *p*. Observe that the stronger antisymmetry allows to infer identity between two places *p* and *q* when *p* is contained in *q* and vice versa.

Orientation. Finally, we need some relations to ensure *orientation* and *direction* in space giving an account of entities' disposition: this can be achieved assuming particular reference points. Assuming specific reference points consists in ordering entities with respect to these particular points, that is, in such a way that for every entity the relation with the reference point is stated directly or indirectly (if related to another entity related to that point). Besides the contingent choice of the reference point, what is really important is the notion of *order* coming from the existence of the reference point. Of course, Cardinal Points can provide a natural choice for orienting in a 2D space; as a result, the four orientation relations R_N, R_E, R_S and R_W would be introduced (where a relation $R_N(p,q)$ holds if *q* is *north of p*). Formally, orientation relations are *irreflexive*, *asymmetric* and *transitive* relations, that is, they are *strict partial orders* on the set of places; the order is "partial" because two places may be incomparable, and a greatest element always exists, namely, the top of the relation (e.g. North for R_N). Some relations can be defined as the converse of other ones (e.g. R_S of R_N), and other non-primitive relations such as *north-east of* (R_{NE}), can eventually be defined by means of usual set theoretic operators from the previous ones, e.g. $R_{NE} = R_N \cap R_E$. Observe that, following this approach, different orientation relations can hold between two places.

With respect to the example of Fig. 6.2, the generation of the spatial model is represented in the right-most square: the nodes are the interesting places (rooms and sensors), while proximity and containment relations are represented by dashed and unbroken lines respectively. Orientation relations can be derived from the compass icon, but have been omitted in the figure.

The classes of proximity, containment and orientation relations are particularly relevant to commonsense spatial reasoning; in fact, the joint assumption of relations of the three classes, although not provably exhaustive, provides a characterization of the environment which meets at least the tree following representational requirements:

- the definition of a basic graph relating places according to their reachability by means of proximity relations. From our perspective, this gives an account of the notion of connection, although it does not ground it on the spatial internal structure of the involved entities.
- A rough (qualitative) ordering of places in a 2D or 3D space by means of orientation relations (3D if a vertical order is added): this is important to reflect the idea of disposition of places and objects in space. Neither a grid nor a Cartesian reference system is employed here, but the notion of disposition is traced back to the concept of *order*, and more precisely, to the projection of various orders on the place domains.
- The possibility of taking into account places of various types and size, representing different layers of abstraction by means of containment relations (a desk inside a room, a room inside a building).

Since those three classes are so relevant, they can be exploited to identify a particular class of CSMs. The class of *standard CSM* is defined as the set of CSMs whose relations all belong to the categories discussed above (proximity, containment and orientation) and there is at least one relation for each category. Obviously, the set of places must include the reference points of the chosen orientation relations. Let proximity, containment and orientation be three classes of relations for which the formal properties discussed above hold, formally a Standard Commonsense Spatial Model can be defined as follows:

Definition 2. Let assume that $\left\{R_1^p,...,R_k^p\right\}$ is a set of proximity relations, $\left\{R_1^c,...,R_m^c\right\}$ is a set of containment relations, and $\left\{R_1^o,...,R_n^o\right\}$ is a set of orientation relations, each one with its top element top_i. A Standard Commonsense Spatial Model SCSM is a CSM with $R_S\left\{R_1^p,...,R_k^p,R_1^c,...,R_m^c,R_1^o,...,R_n^o\right\}$ and $\left\{top_1,...,top_n\right\} \subseteq P$.

Properties discussed so far refer to relations when they are considered individually. Besides these properties, there are *cross properties* of the model concerning the relations among different CSRs and their interdependencies. For instance, the two relations R_N and R_S should be defined as mutually converse. The logic should be expressive enough to represent also cross properties, and this is actually the case of Hybrid Logic.

6.4 Hybrid logics for commonsense spatial reasoning

CSMs provide a general approach to commonsense spatial representation, and SCSMs present a particular subclass of models narrowing the reasoning scope. The adoption of relational models as a basis for the spatial representation makes modal-like logics, and in particular, hybrid logic, a promising framework to support reasoning.

6.4.1 The hybrid logic approach

Modal logics can be viewed as fragments of first order logic that generally provide a good trade off between expressiveness and acceptable computational behaviours. These logics, originally born to reason about possibility and necessity, developed into a powerful formal framework to reason about relational structures with a particular perspective over reasoning; for a good introduction to Modal Logic from this "modern" perspective refer to Blackburn (2000).

Formulas of modal languages are compositionally built taking propositional formulas, usual logical connectives and modal operators as primitive elements. There are two types of modal operators: diamonds ($\Diamond$) and boxes ($\Box$). In a spatial modal language, one can introduce two operators $\Diamond_{IN}$ and $\Box_{IN}$, meaning intuitively "possibly in" and "necessarily in": a formula such as $\Diamond_{IN}$ *intrusion* means that in some place inside the current place the formula *intrusion* is true; a formula such as $\Box_{IN}$ *intrusion* means that everywhere (in every place) inside a current place the formula *intrusion* is true. Box and diamond operators are strongly connected since $\Box\varphi$ is equivalent to $\neg\Diamond\neg\varphi$.

As for the semantics, in modal formulas are usually evaluated over nodes of relational structures. A valuation function assigns to propositional formulas the states of the model in which they are satisfied; the interpretation of complex formulas is defined recursively as usual for logical connectives. The interpretation of a modal operators is tightly connected a relation of the model. Such operators allow to explore the model following the edges of the relational structure corresponding to operators' relations. The spatial interpretation of modal languages consider modal operators as spatial operators, whose semantics is defined by the respective Commonsense Spatial Relations; formulas are evaluated over places with the operators allowing the spatial exploration of the CSMs. The particular modal perspective over reasoning is based on the concept of "locality": a formula such as $\Diamond_{IN}$*intrusion* may be satisfied in a place and not in another one. Truth and meaning of spatial modal formulas depend therefore on the spatial context in which they are evaluated.

Hybrid Logic adds to the modal perspective features that are particularly useful with respect to our approach to commonsense spatial reasoning. In fact, hybrid languages are modal languages providing constructs to refer to specific states of the model and to express sentences about satisfiability of formulas in the language itself. Therefore, it is possible to reason about what is going on at a particular place and to reason about place equality (i.e. reasoning tasks that are not provided by basic modal logics). Formally, this is achieved by adding to the language a specific set of propositional symbols *NOM*, called “nominals”, disjoint from the set of usual propositional symbols *PROP*, and introducing a set of “satisfaction operators” $@_i$. Nominals are similar to propositional symbols (they can be used in the same way) but they are evaluated to be true at a single state of the model, which is called its *denotation*, referring to unique states in the model. Therefore, assume that i is a nominal, $@_i$ a satisfaction operator, and $@_i \varphi$ a formula; wherever $@_i \varphi$ is evaluated, φ must be satisfied at the state of the model denoting i. Formally, semantics is given as usual for modal logics, adding the following truth condition statements for the set of satisfaction operators:

$$W\ , w \models @_i \varphi \text{ if and only if } W\ , w' \models \varphi,$$

where the place w' is the denotation of i, i.e. $V(i) = w'$.

6.4.2 Hybrid commonsense spatial reasoning

Every CSM, being a relational structure, can be taken as a *frame* (and, thus, classes of CSMs such as the SCSMs identify classes of frames), in which the states of the model are places and the CSM relations specify the meaning of the spatial modal operators that can be chosen (thus, from now on we use *places* to refer to *states of the model*). A Hybrid Commonsense Spatial Model is thus given by a CSM and an evaluation that provides a recursive interpretation of formulas of a hybrid language: the base step specifies which propositional variables are true in which place and which is the denotation of the nominals introduced. Formulas of the language allow to represent information spatially referenced, and in particular propositional variables coding particular atomic information are spatially referenced according to the evaluation given by the interpretation function.

As an example, the following spatial modal hybrid language can be interpreted over Standard Commonsense Spatial Models.

Definition 3. Basic Standard Commonsense Spatial Language ($\Lambda^{SCSMbasic}$). $\Lambda^{SCSMbasic}$ is a hybrid language containing the modal operators $\Diamond_N$, $\Diamond_E$, $\Diamond_S$, $\Diamond_W$, $\Diamond_A$, $\Diamond_{IN}$, $\Diamond_{NI}$, and where $\{north, east, south, west\} \in NOM$.

The formal semantics for the $\Lambda^{SCSMbasic}$ language is defined accordingly to the definition of Standard CSMs, and can be found in (Bandini and Palmonari, 2005b) with a complete axiomatization. The intuitive meaning of $\Diamond_N$ is “possibly north of” with $\Diamond_N \varphi$ meaning that there is some place north of the current one in which φ is true; $\Diamond_A$ stands for “possibly proximal to”; $\Diamond_{IN} \varphi$ means that there exist a place contained in the current one, in which φ is true; conversely, its inverse $\Diamond_{NI}\varphi$ is satisfied when φ is true in a place that contains the current one.

The combination of the multimodal and hybrid expressiveness provides a powerful logical reasoning tool to shift perspective on a specific place by means of a $@_i$ operator, which allows checking properties holding over there; for instance, with respect to Fig. 6.2, the information "a glass is broken" referred to window's sensor can be represented in the hybrid language with the formula $@_{window\_sensor} broken\_glass$. Then, operators provide insights on spatial relations; for example, if one wants to query if *in* some room *at west of* the kitchen some *smoke* has been detected, then he/she should query if the formula $@_{kitchen}\Diamond_W \Diamond_{IN} broken\_glass$ is satisfied in the model.

The key element to enable spatial inferences consists in the possibility to define logical properties of spatial relations within the language. From the hybrid logic perspective, this means to be able to define the frame specifying the modal axioms characterizing the formal meaning of the modal operators involved. Hybrid Logic's expressiveness allows to define a class of formal properties of relations much larger than plain modal logics; in particular, all the properties involved in the characterization of SCSMs can be asserted. From a formal point of view, this means that a large class of frames can be defined in a very intuitive way, and therefore this feature of Hybrid Logic has been addressed as *frame definability* in Bandini et al. (2005c). Table 6.1 shows hybrid formulas defining the main properties of spatial relations involved in the definition of Standard Commonsense Spatial Models.

There is a further issue relevant to our purposes, though. Hybrid Logic allows us to take advantage from *frame definability* for reasoning. In fact, Blackburn (2000) proved that if a frame is defined by a set of formulas, it is always possible to build a logical calculus that is sound and complete with respect to that frame. The calculus is built adding to a tableaux system the formulas defining the operators' meaning; we refer to this property as *modularity*.

Table 6.1. Definability of properties for SCSMs

	Property	CSR class	Definition
(*ref*)	*reflexivity*	P,C	$@_i \Diamond i$
(*irref*)	*irreflexivity*	O	$@_i \neg\Diamond i$
(*sym*)	*symmetry*	P	$@_i \Diamond\Diamond i$
(*asym*)	*asymmetry*	O	$@_i \neg\Diamond\Diamond i$
(*antisym*)	*antisymmetry*	C	$@_i \Box(\Diamond \rightarrow i)$
(*trans*)	*transitivity*	C,O	$\Diamond\Diamond i \rightarrow \Diamond i$

Frame definability and modularity make Hybrid Logic a highly desirable candidate for spatial reasoning with respect to our correlation purposes. A first result is the following: as shown in Bandini et al. (2005c), for every Standard Commonsense Spatial Model there exists a sound and complete tableau based calculus that can be easily built (formal details are omitted here but can be found in the reference above). This means that it is possible to exploit all the relations defined in section 6.3.3 within a full logical calculus.

A second but not less important outcome concerns the general strategy defined for dealing with correlation of information in Pervasive Computing domains. It is possible to assume as reference Standard Commonsense Spatial Models where a number of relations are already defined, but relations can be changed or modified in a very flexible way (for classes of frames not definable with hybrid logics without quantification see Blackburn (2000)). This means that the passage from the definition of a relational spatial model to the logic in order to enable reasoning over it is highly simplified and backed by a set of theoretical and formal results (in particular, based on Blackburn (2000) and Bandini et al. (2005c)).

6.4.3 Logical reasoning: inferring scenarios and time

Let us sum up the steps done so far: we provided a class of Commonsense Spatial Models and defined a formal framework to reason over such models exploiting some relevant Hybrid Logic features. Formulas defining frame classes, e.g. SCSMs, allow to perform spatial inference by encoding the high-level properties of the spatial relations.

This logical apparatus can be applied now to reason about correlation as far as formulas defining interesting correlation can be written in the language. In general, events can be represented in the language by propositional formulas and the same can be done for scenarios. Let *s* be a propositional symbol representing a scenario and *preconditions* an arbitrary modal formula representing logical spatial correlations among events. Scenarios can be therefore defined by means of formulas of the form:

$$preconditions \leftrightarrow s$$

Double implications can be used along both the directions: to infer scenarios when preconditions hold, or, conversely, to go checking if the preconditions of a target scenario hold. This general formula schema hides a high flexibility in the scenarios' definition, that results clear just modifying the type of the *preconditions* formula. If *preconditions* is a formula of the form $\varphi_0 \wedge ... \wedge \varphi_n$, then a conjunction of formulas must be considered; disjunctions can be used as well, but it is possible to exploit also modal operators. As an example, consider that *preconditions* is a formula such as $\Diamond_{IN}\varphi$, or $\Diamond_{N}\varphi$; the first formula means that a scenario is defined to hold on a place when preconditions defined by φ hold in some place in it; the second one forces to go checking the preconditions φ for a scenario in places that are north of it. Moreover, correlation can be stratified deriving higher-level scenarios from the correlation of lower-level ones (i.e. *s* become a higher-level scenario if formula representing lower-level scenarios are used in *preconditions*).

An example in the Smart Home domain is presented in the next section. Nevertheless two concerning reasoning have been left open and still need to be addressed. First, considering the temporal dimension in the correlation activity raises mainly computational problems, rather than representational ones. It would not be difficult to add to a modal language some operators to take into account time. Different temporal modal logics have been defined and modal logics have been presented as a promising framework to handle spatio-temporal reasoning by Bennett et al. (2002). As for the scenarios to infer, spatio-temporal scenarios should be taken into account and should be defined on the basis of spatial and temporal correlations among events; the definition of such correlations can be supported by adding suitable temporal operators. An example of spatio-temporal correlation modeled with a spatio-temporal modal logic can be found in Bandini et al. (2005a), which presents a system, i.e. SAMOT, devoted to monitor and control traffic anomalies on a highway. The computational problems arise since spatio-temporal modal logics can be defined as multi-dimensional modal logics, which generally have bad computational properties up to undecidability (Blackburn et al., 2000).

Second, the logical framework presented provides a sound and expressive modeling environment. From this perspective the logical calculus allows to control the intended meaning of the employed spatial relations with respect to inference and deduction. When an efficient implementation is required it is possible to approximate the logical model by means of other knowledge-based computational tools, such as production rule systems. An example is provided by the above mentioned system SAMOT, where the logical model for alarm correlation has been modeled with a spatio-temporal (undecidable) modal logic and therefore implemented into a production rule system.

6.5 A Smart home example

Let us show two examples of correlation with respect to the Smart Environment introduced in section 6.3.2.

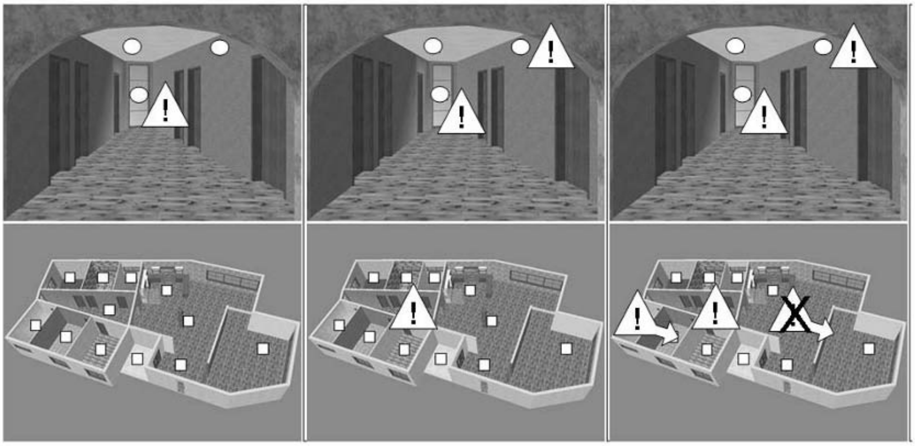


Fig.6.3. Some examples of scenarios deduced from local information

A single alarm is often not sufficient to infer that in a wider zone where it has been sensed, there is a dangerous situation. So, in order to reduce the false alarm rate, it can be required that at least two different devices must return alarm signals. This can be captured by a *scenario* coded, within the $\Lambda^{SCSMbasic}$ language, into a proposition "*Al*2", whose meaning is that a second-level alarm occurs; the latter can be therefore inserted in the consequence of the following formula:

$$@_i \left(Al \wedge \Diamond_{NI} k \right) \rightarrow \left(@_j \left(Al \wedge \Diamond_{NI} k \wedge \neg i \right) \rightarrow @_k \; Al2 \right)$$

The formula states that, if an alarm has been detected in two different places *i* and *j* both contained in another place *k*, it should be inferred that in *k* there is an a second-level alarm. Now, assume that s_1 and s_2 are two sensors that detected an alarm ("*Al*"), and they are both contained in the *corridor* ("*cor*"), with s_1, s_2 and *cor* being all nominals; it can be proved that a *second-level alarm* ("*Al*2") is inferred to occur in the corridor; a proof of this deduction can be found in Bandini et al. (2005c). These two situations are represented in Fig. 6.3: in the left-most figure only one alarm is present in the corridor and the scenario is not inferred; in the central figure a "danger in room" scenario is recognized.

A further example of inference is sketched in the right-most side of Fig. 6.3, and it shows the attrattiveness of the "local" perspective on reasoning provided by modal logics; this example is based on the specification of scenarios that are recognized in a context dependent way, on the basis of the location in which deduction starts. Suppose to define scenarios such as "danger at East", "danger at North", and so on. This can help if we are informed by an alarm notification and we want to know if the alarm is on the way toward the exit. Let us insert the axioms following this schema: $\Diamond_E Al2 \rightarrow e.d.$ (*e.d* is short for East danger). Now suppose to be in the left most room – i.e. *room*1 - and ask if there is an *e.d* scenario. It is provable within the calculus for SCMS that the answer is positive. On the contrary, the same would not hold starting the proof, e.g. from the *kitchen*. In this example, a proper spatial inference is involved in the proof, since the premises do not explicitly state that the corridor is *east* of *room*1 and the transitivity of the orientation relation is exploited.

6.6 Concluding remarks

Although information can be geo-referenced according to quantitative models, spatial representations used for correlation of information coming from distributed sources should rely on stronger semantic spatial references, that is, spatial entities meaningful for the correlation task. From this perspective, Qualitative Spatial Representation and Reasoning offers a framework to enable spatial inferences supporting correlation of information based on a high-level representation of the environment.

A class of Commonsense Spatial Models has been characterized from a formal point of view and some relevant classes of relations have been analysed. Beyond foundational concerns about proper spatial representations, an important issue concerns the relationship among models, languages and, most of all, inferences. Although Commonsense Spatial Models are quite simple, the joint assumption of such models and of a Hybrid Logic approach to reasoning proves to be enough powerful and flexible to reason in such contexts.

Naturally, qualitative spatial reasoning cannot be taken as alternative to well known quantitative spatial models widely used in Pervasive Computing, which generally provide information e.g. about position, and about lot of spatial information with GIS. A major issue for future work, as suggested also by Fonseca et al. (2002), consists instead in the full integration within concrete applications of different levels of representations, and, in particular, of qualitative and quantitative ones.

References

Aiello, M., and J. van Benthem (2002): A Modal Walk Through Space. *Journal of Applied Non-Classical Logics* Vol. 12 No. (3-4), pp. 319-364

Bandini, S., Bogni, D., and Manzoni, S. (2002): Alarm Correlation in Traffic Monitoring and Control Systems: a Knowledge-Based Approach. In: *Proceedings of the 15th European Conference on Artificial Intelligence*, July 21-26 2002, Lyon (F). Amsterdam: IOS Press, pp. 638–642.

Bandini, S., Alessandro M., Palmonari, M., and Sartori, F. (2004): A Conceptual Framework for Monitoring and Control System Development. In *Ubiquitous Mobile Information and Collaboration Systems* (UMICS'04), LNCS, Vol. 3272. Springer-Verlag, pp. 112-125.

Bandini, S., Bogni, D., Manzoni, S., and Mosca, A. (2005a): A ST-Modal Logic Approach to Alarm Correlation in Monitoring and Control of Italian Highways Traffic. In *Proceedings of The 18th International Conference on Industrial & Engineering Applications of Artificial Intelligence & Expert Systems.* Bari, June 22-25, 2005. Springer-Verlag, pp. 819–828.

Bandini, S., Mosca, A., and Palmonari, M. (2005b): Commonsense Spatial Reasoning for Context–Aware Pervasive Systems. *Location- and Context-Awareness, First International Workshop*, LoCA 2005, LNCS Vol. 3479. Springer-Verlag, pp. 180–188.

Bandini, S., Mosca, A., and Palmonari, M. (2005c): A Hybrid Logic for Commonsense Spatial Reasoning. *Proceedings of the 9th Congress of the Italian Association for Artificial Intelligence*. LNAI 3673, Springer-Verlag, pp. 25–37.

Bennett, B. (1996): Modal Logics for Qualitative Spatial Reasoning. *Journal of the Interest Group in Pure and Applied Logic*. Vol. 4, No. 1. pp. 23-45.

Bennett, B., Cohn, A. G., Wolter, F., and Zakharyaschev, M. (2002): Multi-Dimensional Modal Logic as a Framework for Spatio-Temporal Reasoning. *Applied Intelligence* 17 (3), pp. 239–251.

Blackburn, P. (2000): Representation, Reasoning and Realtional Structures: a Hybrid Logic Manifesto. *Logic Journal of the IGPL*, Vol 8 No. 3, pp. 339–365.

Blackburn, P., de Rijke, M., and Venema, Y., (2000): Modal Logic.. Cambridge University Press.

Carvalho, H.S., Heinzelman, W.B., Murphy, A.L., and Coelho, C.J.N. (2003): A General Data Fusion Architecture. *Proceedings of the 6th International Conference on Information Fusion (Fusion)*. Cains, Australia, pp. 1465- 1472.

Casati, R., and Varzi, A. (1999): Parts and places: the structures of spatial representation. Cambridge, MA and London: MIT Press.

Chen, H., Finin, T., and Joshi, A. (2004): An Ontology for Context-Aware Pervasive Computing Environments. *Knowledge Engineering Review*, Vol. 18 No. 3, pp. 197–207.

Christopoulou, E., Goumoupoulos, C., and Kameas, A. (2005): An ontology-based context management and reasoning process for UbiComp applications. In *Joint conference on Smart Objects and Ambient Intelligence (sOc-EUSAI)*. ACM Press, pp. 183–188.

Cohn, A. G, and Hazarika, S. M. (2001): Qualitative Spatial Representation and Reasoning: An Overview. *Fundamenta Informaticae*, Vol. 46 No. (1-2):, pp.1–29.

Dey, A. K., (2001). Understanding and Using Context. *Personal and Ubiquitous Computing*, Vol. 5 No. 1, pp. 4–7.

Egenhofer, M. J., and Mark, D. M. (1995): Naive Geography. In *Spatial Information Theory - A Theoretical Basis for GIS (COSIT'95)*, edited by A. U. Frank and W. Kuhn, Berlin, Heidelberg: Springer, pp. 1–15.

Fonseca, F. T., Max, J., Egenhofer, C., Davis, A., and Câmara. G. (2002): Semantic Granularity in Ontology-Driven Geographic Information Systems. *Annals of. Mathematics and Artificial Intelligence*, Vol. 36 No. (1-2), pp. 121–151.

Ossowski, S., Hernández, J., Belmonte, M., Maseda, J., Fernandez, A., García-Serrano, A., Triguero Ruiz, F., Serrano, J. M., and Pérez de-la Cruz, J. L. (2004): Multi-Agent Systems For Decision Support: A Case Study In The Transportation Management Domain. *Applied Artificial Intelligence*, Vol. 18 No. (9-10), pp. 779–795.

Randell, D. A., Cui, Z., and Cohn, A. G. (1992): A Spatial Logic Based on Regions and Connection. *Proceedings of the 3rd Internationa Conference on Knowledge Representation and Reasoning*. San Mateo, CA: Morgan Kaufmann, pp. 165–176.

Zambonelli, F., and Parunak, H.V.D. (2002): Signs of a Revolution in Computer Science and Software Engineering. *Proceedings of Engineering Societies in the Agents World III* (ESAW2002), Volume 2577. Springer-Verlag, pp. 13–28.

7 Personalising Map Feature Content for Mobile Map Users

Joe WEAKLIAM[1], David WILSON[2], Michela BERTOLOTTO[1]

[1]School of Computer Science and Informatics, University College Dublin, Belfield, Dublin 4, Ireland
[2]Department of Software and Information Systems, University of North Carolina at Charlotte, NC 28223, USA

Abstract. Several challenges arise when displaying maps on mobile devices. Users encounter problems dealing with information overload and mapping interface interaction while on the move – issues of human-computer interaction. In addition, to effectively display maps on mobile devices, developers must address restrictions related to screen size and limited bandwidth – issues of computational efficiency. We have developed MAPPER, a novel approach for delivering personalised maps to mobile devices, which addresses mobile mapping issues from both perspectives. MAPPER generates maps containing specific spatial feature content that is tailored to the explicit preferences of users with contrasting requirements by monitoring the interactions of individuals when browsing maps. All interactions between users and maps are captured implicitly and are used to infer individual and group preferences related to specific map feature content. MAPPER provides an effective and efficient means of delivering and representing maps on mobile devices, which addresses information overload by providing exactly the map information necessary to suit user interaction preferences. In turn, tailoring map content to user preferences considerably reduces the size of vector datasets necessary to transmit and render maps. This chapter describes the map personalisation approach in MAPPER and presents a user study showing the benefits of providing a diverse set of individuals with personalised map feature content when engaged in mobile mapping tasks.

7.1 Introduction

Interaction with mobile applications is constrained by several well-known device limitations compelling developers that design mapping systems for mobile devices to address these restrictions in order to produce practical applications. Device restrictions include, amongst others, low bandwidth for transmitting datasets used to render maps, comparatively poor computational power for processing the datasets, and limited screen sizes for displaying the map content described by the datasets. When browsing maps on mobile devices humans can encounter major difficulties related to information overload, common in any IR (Information Retrieval) system, and human computer interaction (HCI). Information overload is an expression used to describe situations that arise when humans are no longer able to effectively process the amount of information they are exposed to whereas HCI problems become prevalent when users try to access specific information using mobile devices when on the move. One solution that addresses the problems associated with mobile device limitations, information overload, and HCI, is to reduce the amount of data necessary to generate maps on the mobile device. The question is how to reduce the size of the dataset in a manner whereby the user still has all the information necessary to complete their current task, i.e. omitting only superfluous detail.

Reducing the size of datasets needed to display maps is relatively easy when users can be solicited for specific input related to their map feature preferences. However, demanding input explicitly from individuals every time they request a map or, worse still, while on the move, is not a practical solution. Therefore, how does one reduce the amount of data needed to render a personalised map without relying on explicit user input? One possibility is to maintain profiles of user map feature preferences so that no burden is placed on the user to provide input whenever they request maps. User models can be based on explicit input (static models), implicit input (dynamic models), or a combination of the two (Fischer, 2001).

Dynamic profiling is the process of analysing a user's activities or actions implicitly to determine what the user is interested in. Maintaining models of user map feature interests implicitly offers a more suitable solution than using static models as at no point is the user solicited for any input. Implicit profiling techniques make use of interaction information captured at the interface and are common in IR applications on the Web. The user simply browses through the information presented looking for content of interest where user actions, along with additional content information, are recorded. Once the user has completed their task, this information is analysed for trends in user behaviour. The concept of dynamic profiling can be adapted to the scenario of users browsing maps. User preferences regarding various aspects of map feature content are added to a user model, which is continuously updated every time the user interacts with maps. The more the user requests and interacts with maps on mobile devices, the more their profile evolves to accurately reflect their specific map feature content preferences.

Creating and maintaining dynamic models of user preferences based on implicitly captured detail enables the size of datasets required to render mobile maps to be reduced considerably, as core user preferences can be established without the need for any explicit user input. This is known as map personalisation. Not only does the significant reduction in the size of the datasets facilitate the display of map feature content on mobile devices by countering device limitations, but it also benefits the user as issues related to information overload and HCI can be addressed. Furthermore, personalising mobile maps allows the user to complete their tasks with greater ease due to the presence of task-relevant feature content and the removal of extraneous content.

In this chapter we introduce MAPPER (MAP PERsonlisation), which describes a novel approach to personalising maps and forms the personalisation component of the CoMPASS system (Weakliam et al, 2005b). Personalisation with MAPPER is provided at two distinct levels. The first is personalisation at the layer level whereby the system recommends a subset of spatial map layers from a dataset describing all possible map layers, and then for each recommended layer all the features (the entire dataset) representing that particular layer are displayed. An example of this might be presenting a user whose preferences centre on hydrographical content with a map displaying all the rivers, lakes, reservoirs, canals, etc. for the map region requested and omitting extraneous layers, e.g. golf courses, hiking trails, and airports, as the user expressed no previous interest in these layers. Alternatively, map personalisation can be provided at an even higher level of detail, i.e. at an individual feature level. This means not only generating maps containing a subset of the total set of map layers available, but also personalising the content described by the individual map layers, i.e. personalisation at a finer grained feature level. Taking the above example would result in presenting the same user with a map containing largely hydrographical layers and for the highest relevance layers returned, displaying only a fraction of the total dataset that represents the layer based on the user's preferences, i.e. only displaying lakes with an area greater than a specified threshold. It should be noted that other types of personalisation are also possible, e.g. map region and map scale personalisation, but we do not address them here. We focus on providing map content personalisation at both the map layer and feature level in MAPPER.

This chapter is structured as follows. Section 7.2 provides a discussion on relevant research in the area of user modelling, map-based mobile services, and interaction in mobile environments. Our approach to mobile map personalisation is outlined in section 7.3 with details about an implemented mobile map personalisation application called MAPPER described in section 7.4. An evaluation of our application along with results is given in section 7.5. Section 7.6 concludes and proposes future extensions to the research described in the chapter.

7.2 Related work

User modelling enables system developers to represent and reason about the interests or preferences of users. There are many different techniques available when modelling user preferences. These approaches can be divided into two categories: those that are reliant on users providing explicit input (static models) and those that profile user preferences implicitly (dynamic models). Several problems arise when utilizing static user models to store user preference detail. Systems that solicit their clients for input place a burden on the user to input demographic detail and information describing their preferences explicitly. Moreover, static profiles degrade in quality over time and the explicit input provided is based on the individual's interest and may not accurately reflect an objective view that can infer the interests of other users with similar interests. As a result there has been quite a notable shift in focus in recent years to implicit methods for profiling user preferences.

The notion of using a "user" and "expert" model to individualise the selection of instructional topics is described in (Linton et al, 1999). The new user model is based on implicitly observing the individual's behavior in a natural environment over a long period of time, while the expert model is based on pooling the knowledge of numerous individuals. Individualized instructional topics are then selected by comparing an individual's knowledge to the pooled knowledge of his peers. Employing Bayesian user models to infer a user's needs by considering a user's background, actions, and queries, is reviewed in the Lumiere Project (Horvitz et al, 1998). Several problems are addressed in the Lumiere project including: (1) the construction of Bayesian models for reasoning about the time-varying goals of computer users from their observed actions and queries, (2) gaining access to a stream of events from software applications, and (3) developing persistent profiles to capture changes in a user's expertise. Both (Linton et al, 1999) and (Horvitz et al, 1998) propose the development of detailed models of user preferences based on implicitly observing the long-term usage of an application. When personalising mobile maps, MAPPER uses similar techniques to model user preferences in map feature content. All interactions between users and maps are recorded implicitly over long periods of time and are then used to infer user interest in various aspects of spatial map content.

Implicit feedback techniques are commonly used for query expansion and user profiling in information retrieval (Kelly and Teevan, 2003). Some typical user behaviors that have been extensively investigated as sources of implicit feedback include reading time, saving, printing, and selecting. Implicit measures are generally thought to be less accurate than explicit measures but as large quantities of implicit data can be gathered at no extra cost to the user they offer attractive alternatives. In (Kelly and Belkin, 2001) three specific hypotheses related to implicit measures of user interest are tested: (1) users will spend more time reading documents they find relevant, (2) users will scroll more often within documents they find relevant, and (3) users will interact more with documents they find relevant. A similar approach is described in (Kim et al, 2001) where the following hypotheses are tested: (1) on average users spend more time reading relevant full-text journal articles than non-relevant articles and (2) combining reading time and printing behavior will be more useful for predicting explicit ratings than using reading time alone. Similar hypotheses within the realm of map browsing and interaction are tested in our previous work (Weakliam et al, 2005a). In particular the following two hypotheses are tested: (1) users will interact more with map features they find relevant and (2) users will spend more time interacting with map regions containing content they find relevant. Details and results of experiments carried out to test these hypotheses are described in (Weakliam et al, 2005a) and indicate that these two measures are useful sources of implicit feedback for ascertaining user interest in map feature content. MAPPER employs both these hypotheses when establishing user preferences related to map feature content.

Many mapping applications utilize user modelling as a means of profiling user interests in various aspects of spatial and non-spatial information (Fink and Kobsa, 2002). The dynamic generation of tourist maps is proposed in (Zipf, 2002) where maps are generated based on many variables ranging from user preferences, the user task, and cultural aspects, to communicative goals and actual context and location. The focus is on personalising maps according to user-specific constraints. These constraints include: (1) the nationality of the user which defines the look of all available features ensuring that the user gets a map in a design that they are familiar with, (2) the current user task which determines what features need to be displayed, (3) the interests of the user which also determines what features need to be displayed, and (4) the layer and scale styles which dictate how the features appear at different map scales. Although information overload is adequately addressed, all input related to the user model and constraints, i.e. demographic detail, the user's task, preference information, etc., must be provided explicitly by the user before any personalised maps are generated. MAPPER, on the other hand, is not reliant on explicit input and makes use of implicit user input only.

CRUMPET (Schmidt-Belz et al, 2002) is a mobile GIS that proposes the adaptation of content to the user's individual interests through user modelling techniques that dynamically profile personal preferences. Like MAPPER, CRUMPET learns user preferences over time, where user profiles are automatically updated based on user interaction history, and the interests of single users as well as groups are considered when personalising maps. If a user asks for more information about an object, CRUMPET makes the assumption that the user is interested in objects with these features more so than in others. This is similar to assumptions made by MAPPER, e.g. if a user highlights a region containing lakes than that user is interested in lakes in general. However, in CRUMPET some services must be requested explicitly in order for the user model to adapt to user interests, whereas user models in MAPPER adapt to user preferences based solely on information gathered implicitly. Moreover, CRUMPET has been developed largely for the tourist market, while MAPPER can generate personalised maps for users with any combination of content preferences.

A system that delivers various types of information to mobile devices based on location, time, and profile of end users is outlined in (Hinze and Voisard, 2003). Personal profiles are defined explicitly by the end user or are inferred from user actions applying user-profiling techniques. Although automatically inferring profiles based on user interactions is proposed, no details about the user-profiling techniques are mentioned nor are there results to suggest that implicitly profiling user preferences benefits the user when using the application in question. FLAME2008 (Weisenberg et al, 2004) is a situation-aware GIS that delivers personalised Web services to its users. However, unlike MAPPER, FLAME2008 relies on users providing profile information, i.e. demographic details and preference information, explicitly.

The goals and practices of a nomadic art exhibition guide called Hippie are described in (Oppermann and Specht, 2000). Each user of Hippie is associated with a dynamic user model describing the knowledge, interests, movements, and personal preferences of the user. Positive evidence for gaining knowledge about exhibition objects comes from users listening or watching presentations, whereas negative evidence comes from users skipping or stopping presentations. A tour encompassing exhibits of specific genres is then recommended dynamically by the system. Results show that the provision of personalised views and individual tours, together with adaptive information selection, reduces redundancy and information overload. In contrast to Hippie, the focus of MAPPER is on the dynamic recommendation of specific map feature content to assist users when completing mapping tasks while on the move.

A general conceptual framework for mobile cartography is described in (Reichenbacher, 2001a) and (Reichenbacher, 2001b). Stereotypes, where further subcategories can be derived, are used to represent user preferences and to filter information, resulting not only in a reduction in the amount of information presented to users, but also in a recommendation of relevant, detailed, and hence appropriate spatial information to users. A distinction is made between different user groups for GIS in a mobile environment as a first step to providing personalisation, e.g. professionals, naïve users, tourists, and everyday users. These user groups differ in their knowledge, information needs, quality requirements, skills, tasks, and so on. Despite the fact that personalised mobile maps containing specific content are generated, the onus is on the user to provide preference detail to the system explicitly whereas MAPPER profiles user preferences implicitly.

7.3 Mobile map personalisation with MAPPER

This section introduces and describes the MAPPER mobile map personalisation application. It is important to note that MAPPER is designed for use with vector data as opposed to raster data to render mobile maps.

7.3.1 Generating and delivering mobile maps

When users request maps while on the move, the user's current task is crucial. It is essential to minimize the amount of map feature data transmitted to the mobile device but at the same time tailor the map feature content to the preferences of the individual requesting the map so that they can complete their task with as little interaction with the mobile application as possible. However, personalising maps is based on establishing knowledge about user preferences. Where does this knowledge come from? Detail describing user preferences comes from the users themselves and is based on implicitly capturing and analysing user browsing behaviour with maps.

The majority of GIS applications that render maps generate static raster maps for their clients (MapQuest, 2006) (Yahoo! Maps, 2006). Raster maps are difficult to personalise, as the map feature content cannot be divided into distinct map layers. In certain cases additional layers can be superimposed onto the original raster map thus allowing certain aspects of the map to be highlighted. However, this does not by any means reduce or personalise the dataset. Using vector data to render maps has several advantages over using raster data: (1) vector data allows for a greater level of interactivity, (2) unlike vector data, it is difficult to reduce the cost of transmission of raster data, and (3) vector data enables maps to be divided into distinct map layers where each layer represents a specific feature on the map, e.g. a lakes layer. With vector data, map layers are either landmark (point/polygon) layers or non-landmark (line) layers. Landmark layers generally form the focus of user sessions, i.e. the user is trying to locate the nearest airport, whereas non-landmark layers, e.g. road network, are typically used for navigating maps. Landmark layers can be further decomposed into sets of individual layer features, i.e. individual lakes vs. all lakes, whereas non-landmark layers are treated as entire layers. Breaking down map layers into sets of individual features facilitates mobile map personalisation at the feature level. Rendering maps using vector data where different map layers are distinguished and access to individual layer features within a particular map layer is possible, allows for detailed information to be captured whenever a user executes even the most simple map action.

We have developed an approach to map personalisation for mobile map applications in the MAPPER system. New users to MAPPER are presented with default maps displaying default feature content. This is because no user preference information has yet been determined. However, once this user interacts with the map, a profile is created for the user so that upon session termination, content preference information extracted from that session is inserted into their profile. Therefore, the next time the user requests a map, specific feature content is recommended to the user based on interaction behaviour recorded from their opening session. Alternatively, if the user has made previous requests, then personalised map content is presented to the user in the form of a personalised map. The more a user interacts with maps, the more detailed the user profile, and hence the more personalised the maps generated (for more information on the structure of the user profile see Weakliam et al, 2005a).

7.3.2 Recording interaction between users and mobile maps

Information about user preferences regarding map feature content must be inferred before personalised maps can be generated. Determining user preferences in aspects of map feature content is based entirely on gathering and analysing data from user-map interactions implicitly. Most applications that produce maps provide basic functionality to their clients for interacting with maps. Standard browsing actions provided include panning, zooming, toggling features on/off, and highlighting map features.

Table. 7.1. Map browsing actions

Action	Type	Layers on which action is executed	Assumption made about user interest in layers/features based on action executed
Automatic zoom in	Weak	Layers in resulting map frame	User may be interested in layers/features in the resulting frame as they zoomed in on an area containing them.
Manual zoom in	Strong	Layers in resulting map frame	User is interested in layers/features in the resulting frame as they explicitly selected a region of the map to zoom in on.
Automatic zoom out	Weak	Layers in resulting map frame	User expresses no definite interest in any layers/features in the resulting frame as they zoomed out to view more content.
Automatic pan	Weak	Layers in resulting map frame	Reveals no evidence of interest in any layer/feature content and generally executed as a means of browsing the map.
Manual pan	Weak	Layers in resulting map frame	Reveals no evidence of interest in any layer/feature content and generally executed as a means of browsing the map.
Toggle map layer on	Strong	Layer toggled on	User expresses explicit interest in the map layer as they toggle it on for the entire map.
Toggle map layer off	Strong	Layer toggled off	User expresses a lack of interest in this map layer as they remove it from the map.

Highlight map layer	Strong	Layer highlighted	User expresses explicit interest in the map layer as they highlight it for the entire map.

When interacting with maps, individuals may execute any number and any combination of map actions to complete their session task. A user session can be divided into a sequence of map frames where each frame is generated as a result of the execution of a particular map action. A map frame represents the visible state of the map at a specific moment in time. Map frames can contain entire map layers, i.e. all lakes for a dataset are contained within the frame, or subsets of map layers. Table 7.1 outlines the various map browsing actions available to users when completing mapping tasks and assumptions made about user interest in map layers/features as a result of the execution of those actions.

Actions are divided into *strong* and *weak* actions. Strong actions are those that generally disclose explicit information about user interest in map content whereas it is more difficult to determine user interests from weak actions. Analysing sequences of map actions executed by users, together with map content detail recorded along with the actions, reveals trends in how users interact with different map layers/features. Using vector maps allows the following detail to be recorded for every user-map interaction during a map session (see section 7.4.2):

- **Action executed:** This allows us to ascertain the significance of both frames generated by actions and map layers/features at the centre of the actions, as some actions are deemed more relevant than others, e.g. highlighting a layer in the map vs. automatic panning of the map (Table 7.1).
- **Map content upon which action was executed:** Using vector data allows detailed spatial information regarding map layers/features upon which actions were executed to be recorded, e.g. the user manually zoomed in on a region of the map containing various parks and lakes. Running specific spatial queries allows this level of detail to be captured.
- **Time of action execution:** Recording the times at which various actions were executed enables the most relevant session map frames to be established. An assumption is made that the longer the time period between two successive actions the more relevant the content present in the frame generated from the first action.
- **Boundary of map frame generated by action:** Recording the boundaries of map frames generated by actions has two benefits. First, using the frame boundary, in conjunction with the time that the frame was generated (time of action execution), allows for a frame score to be calculated for a frame. Second, the smaller the area covered by the frame, the more the user has zoomed in on the map and hence trends in regions and layers/features of interest can be extracted based on these "small" frames.
- **Map layers/features intersecting resulting frame:** Having recorded the frame boundary, it is then possible to ascertain which map layers/features intersect the frame. This entails the generation of specific spatial queries that are sent to a spatial database in order to retrieve all the layers and/or individual features that either lie wholly inside or intersect the actual boundary of the frame. This is extremely significant as patterns in layer/feature presence can then be determined over a period of a number of map sessions.

7.3.3 Acquiring information on user preferences

All the information captured for each action is recorded in log files on the mobile device. Once the session is terminated, the log files are transmitted from the client to the server for detailed analysis. All log file analysis is done offline at the server, thus placing no onus on the mobile device to process any information other than displaying maps. This is essential, as mobile devices in general have restricted processing power. Relevant detail related to map layers, individual map features, and regions of the map are extracted from the log files and inserted into a user profile storing map content preference information. The user profile is then used to generate personalised maps that tailor specific feature content to individuals or groups of users.

7.3.3.1 Determining preferences at the map layer level

Map content detail linked to every single map action executed by users at the interface is captured in log files. What this means essentially is that layer/feature data related to every single map frame generated during every single session is recorded. Capturing this level of detail over large numbers of user sessions provides the basis for ascertaining user interest in content at the level of map layers, i.e. entire sets of features. However, as more and more user sessions are completed, the amount of session information stored in the user profile increases at a massive rate. Therefore, efficient methods for establishing user interest in specific map layers must be established.

Association rule mining, e.g. (Agrawal et al, 1993), is a data mining technique commonly used for determining trends in market basket data. It is possible to apply the concept of association rule mining to the area of map layer presence in session map frames. We use association rule mining for establishing trends in how users interact with map layers and for grouping users with similar content interests together. Similarity measures between pairs of map layers can be calculated based on layer presence in session map frames. Map layers that are highly similar are inclined to appear in frames together, i.e. layer A and layer B are “similar” if frames containing layer A also contain B. In (1) the similarity measure sim_{AB} between two layers A and B is calculated as the Manhattan distance between them.

$$(1)\ sim_{AB} = \left(1 - \frac{f(A \cup B)}{f(A)}\right) + \left(1 - \frac{f(A \cup B)}{f(B)}\right)$$

In (1) *f(A U B)* is the frequency of map frames containing both layer A and B whereas *f(A)* is the frequency of map frames containing layer A. We can also categorize map layers based on similarity scores calculated between specific pairs of layers, i.e. if layer A is similar to layer B, and layer B is similar to layer C, then layer A is similar to layer C as the operation is transitive. Table 7.2 shows a simple user-map session displaying layer absence and presence for each session frame. A ‘✓’ in the map layer column indicates that the layer was present in the frame otherwise the layer was absent from the frame. Recording layer presence in this manner allows us to calculate the Manhattan distances between different pairs of layers so as to determine what layers different users tend to interact with together. This allows map personalisation to be provided at the map layer level and is useful for personalising non-landmark layers, which in turn allows mobile device limitations to be addressed.

Table. 7.2. Simple user session

Frame	Highways	Parks	Rivers	Shops	Hospitals	Local Streets
1	✓		✓	✓		✓
2	✓		✓	✓	✓	✓
3	✓	✓	✓	✓	✓	✓
4	✓	✓		✓	✓	✓
5		✓		✓		✓

7.3.3.2 Using interest frames to determine preferences at feature level

An interest frame is a session map frame that satisfies the following criteria: (a) the time interval between the time at which the frame and the immediate succeeding frame were generated exceeds a threshold value (indicating interest) and (b) the frame was generated as a result of a *strong* map action (see Table 7.1). They are crucial in determining which individual landmark features form the focus of a particular map session. Each interest frame has an associated frame score calculated as follows:

$$(2)\quad frame\_score = frame\_area * num\_frame\_layers * \left(\frac{1}{frame\_time}\right)$$

In (2) *frame_area* denotes the area of the frame, *num_frame_layers* is the number of distinct map layers intersecting the frame boundary, and *frame_time* is the time interval before the immediate succeeding frame is generated. The smaller *frame_area*, the smaller *num_frame_layers,* and the larger *frame_time,* the lower the *frame_score* and hence the more relevant the interest frame. Associating interest map frames with frame scores allows for a hierarchy of map frames to be established for any particular user-map session. This in turn enables the most significant session information to be extracted from the highest relevance interest frames.

Once all the interest frames have been ascertained, the main focus of a user session, as regards trends in specific landmark feature presence, can then be determined through a detailed analysis of the content contained in the interest frames. Individual landmark features are associated with feature scores that are incremented each time the feature appears in an interest frame. Using this simple scoring mechanism enables ratings to be assigned to all unique features present in at least one interest frame during a user session. Amalgamating detail from all sessions involving one specific user means that all individual landmark features can be rated for the user over the duration of any time period allowing for an extremely thorough model of persistent user preferences to be maintained.

7.3.3.3 Modelling long-term user preferences at the layer level

User session information is inserted into the user profile as soon as the user completes their task and terminates the connection to the server. All recorded information in the log files is first analysed at the server before any updates are made to the user profile. It is important to distinguish between long-term and short-term user preferences. Long-term preferences are linked to non-landmark layers and typically represent the map content that users would like present in all map sessions. Long-term content preferences tend to be related to map layers like road features, allowing the user to navigate between points of interest (landmark features). Non-landmark layers are personalised at the layer level, i.e. entire sets of features are returned as opposed to subsets of features. Using association rule mining, similarity scores between different non-landmark layers are calculated offline on a regular basis. This enables groups of relevant and up-to-date long-term layer preferences to be established and hence recommended each time the user requests a map. Therefore, persistent long-term content preferences can be modelled for both individuals and groups of users.

7.3.3.4 Modelling short-term preferences at the layer and feature levels

In contrast to long-term preferences, short-term content preferences are associated with landmark layers and features and tend to be the focus of user sessions, e.g. the user wants to locate the nearest shopping centre to their current location. Short-term preferences are modelled first at the layer level and then at the individual feature level (if the user requests a map of a region that was requested on previous occasions). Landmark layers and landmark features are both scored for every interest frame they appear in. The benefits of this are twofold. First, a hierarchy of landmark layers can be established, thus allowing only a fraction of all landmark layers to be recommended to users, i.e. personalisation at the map layer level. Second, a hierarchy of individual landmark features can also be created within each landmark layer. This facilitates the generation of mobile maps that are personalised at the individual feature level, as not all features of a particular layer type may be returned when a user requests a map, e.g. returning only the lakes that fall within a five kilometre radius of the user's current position. Therefore, the amount of map content used to render a fully detailed map is reduced in two steps, thus benefiting the user largely due to the removal of irrelevant feature content.

7.3.3.5 Personalisation of long-term content preferences

When generating mobile maps, personalisation is provided at two distinct levels – at the layer level and at the feature level. Pertinent non-landmark map layers are personalised at the layer level. When a user requests a map for the first time, i.e. no profile exists in the database, the system recommends only those non-landmark layers that describe the main road network, i.e. interstates, highways, local streets, etc. This is based on an assumption that the majority of users will not be interested in other non-landmark layers and simply require road features to navigate the map. If the user then explicitly requests any other non-landmark layers (rivers, hiking trails, alleyways, etc.) at any time, then this is recorded in their user profile and taken into consideration the next time that user requests a map. The six highest-ranking non-landmark map layers, based on Manhattan distance calculations between pairs of map layers, are presented to the user. As soon as the user terminates a session, association rule mining is run on all session information recorded to date so that all non-landmark map layers can then be reordered if necessary.

7.3.3.6 Personalisation of short-term content preferences

When providing mobile maps with feature level personalisation, only landmark features with a score exceeding a threshold value δ and that belong to landmark layers at the top of the landmark layer hierarchy are returned. It is important to note that if a user requests a map of a region that they have not requested before, short-term content preferences are then personalized only at the layer level based on preferences established from interaction with maps describing other regions. Currently δ is set to 0.25 times the total number of interest frames extracted for all sessions involving the current user to date. What this means is that all landmark features that occurred in at least one quarter of all interest frames for that user and for that map region will be presented to the user. If δ is increased the level of feature personlisation is increased, thus reducing the size of the dataset further, whereas if δ is decreased, more individual features will have a score exceeding δ, thus resulting in more detailed maps. As all landmark layers undergo two personalisation processes, the reduction in the amount of feature content in the mobile maps is considerable. Providing mobile map personalisation in this manner, where the user's attention is immediately drawn to the more detailed parts of the map, is similar to the idea of using focus maps (Zipf and Richter, 2002).

7.4 Designing and implementing MAPPER

This section outlines the implementation of MAPPER. Fig. 7.1 shows the system design detailing how personalised mobile maps are generated.

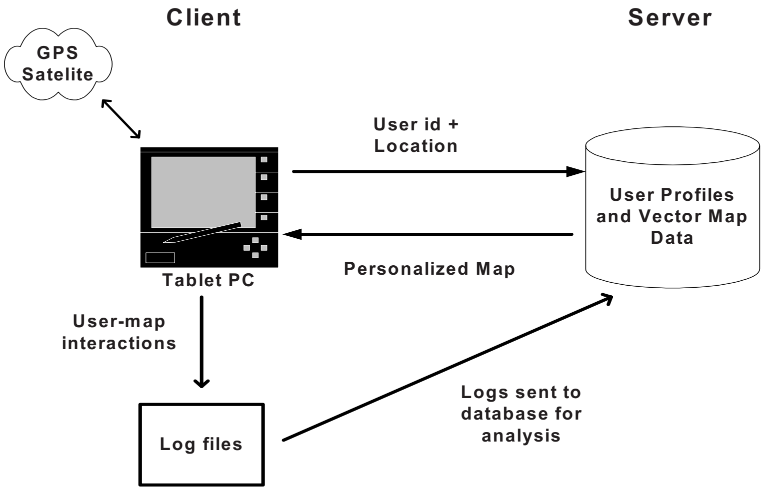


Fig. 7.1. System design

7.4.1 MAPPER Interface

MAPPER has been developed on a Tablet PC. Vector maps are rendered using OpenMap™ (OpenMap, 2006) where the data comes from Tiger/Line 2000 (Tiger,

2006) files that have been loaded into an Oracle 9i spatial database (Oracle Spatial, 2006). Using vector data allows the map to be divided into distinct layers, where each layer can be further decomposed into individual features. The user has the freedom of browsing mobile maps by executing any of the map actions described in Table 7.1. Looking at Fig. 7.2 we can see the different components of the MAPPER GUI.

In Fig. 7.2 the user is presented with a map containing different layers where each layer is categorized as one of the following types:

- **Full layer** – recommended non-landmark layers and landmark layers. For a landmark layer to be displayed as a full layer, all individual features describing the layer must have a score exceeding the personalisation threshold δ.
- **Partial layer** – recommended landmark layer where only a subset of the individual features describing the layer have a score exceeding δ**.**
- **Empty layer** – any layer that is not recommended by the system or any recommended landmark layer where no individual features describing that layer have a score exceeding δ.

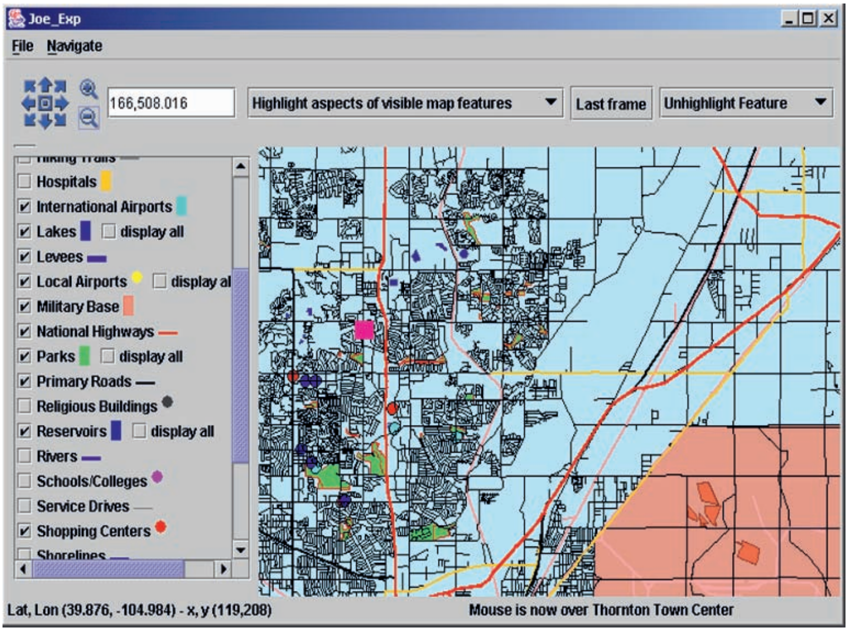


Fig. 7.2. MAPPER application GUI[4]

As is evident from Fig. 7.2, layers that are displayed as partial layers have a second checkbox beside the layer name in the layers panel. This enables the user to request further detail describing the layer if desired. This action is recorded in the log files along with all other map actions and is taken into consideration when updating the

[4] Figures 7.2, 7.5, 7.6, and 7.7 are in color. See accompanying CD-ROM for color versions.

user's profile. If the user wishes to see the names of any features then they simply place the stylus over that feature and the name is displayed at the bottom of the map.

In addition to those actions outlined in Table 7.1 we have also implemented several high level spatial queries allowing the user to highlight various aspects of feature content contained in the map. These queries are classified as strong actions and enable further detail related to user map feature preferences to be ascertained. These may be of interest to professionals requiring access to specific aspects of the spatial map data.

7.4.2 Capturing user-map interactions in log files

All user-map interactions are captured in log files in XML. Using XML facilitates fast parsing of log files and enables specific session information to be extracted from the files once sessions are terminated. Fig. 7.3 shows an excerpt from a sample log file describing the detail that is captured at the map layer level when the user manually zooms in (*z03*) on a specific region of the map. As the detail displayed in Fig. 7.3 is captured only at the layer level, no preference information at an individual feature level, irrespective of whether layers involved in the action are landmark layers or otherwise, can be ascertained through log file analysis.


```
<useraction>
   <mapaction>
      <action_id>z03</action_id>
      <layer_id>D21</layer_id>
      <layer_id>D43</layer_id>
      <layer_id>D61</layer_id>
   </mapaction>
   <frame>
      <frame_number>6</frame_number>
      <frame_time>1115891528609</frame_time>
      <frame_boundary>- 105.0215,39.87568,-
      105.0215,39.84096,-104.96144,39.87568,-
      104.96144,39.84096</frame_boundary>
      <layer_id>D21</layer_id>
      <layer_id>D43</layer_id>
      <layer_id>D61</layer_id>
   </frame>
</useraction>
```


Fig. 7.3. XML excerpt showing map layer level of detail

Fig. 7.4 shows a second excerpt displaying what is recorded at the feature level when a user executes a manual zoom in action. For each landmark map layer that either intersects or lies wholly inside the selected zoom window, the individual features of that layer type that are involved in the action are recorded, e.g. *D43* represents schools shown as points on the map. This allows for more detailed analysis of user interactions as content preferences at the individual feature level can be established.

```
<useraction>
   <mapaction>
      <action_id>z03</action_id>
      <layer_id>D43</layer_id>
      <layer_id>A11</layer_id>
   </mapaction>
   <frame>
      <frame_number>20</frame_number>
      <frame_time>1140795217000</frame_time>
      <frame_boundary>…</frame_boundary>
      <frame_layer>
         <layer_id>D43</layer_id>
         <feature_id>79</feature_id>
         <feature_id>81</feature_id>
      </frame_layer>
      <frame_layer>
         <layer_id>A11</layer_id>
      </frame_layer>
   </frame>
</useraction>
```

Fig. 7.4. XML excerpt showing map feature level of detail

Each user-map interaction results in the generation of a map frame that has several associated attributes, namely a frame time, frame boundary, and frame layers. Interest map frames are extracted from log files based on time and action criteria where a frame score is calculated for each interest frame. If the time interval between two consecutive map frames exceeds a specified threshold *m*, then the first frame is deemed to be an interest frame (*m* is calculated based on each individual user's session history). However, there is also an upper bound on the time interval that elapses between successive frames. If the time interval between two consecutive actions exceeds *k* (60 seconds), then the first frame is not recorded as an interest frame as it is presumed that the user was interrupted in their current task. At the moment we are working with fixed thresholds, as the current focus is to determine whether map personalization can be achieved and if so, does it benefit map users in any way. The next step is to improve the accuracy of the personalization based on each individual MAPPER user, which may involve the incorporation of thresholds with varying values.

7.4.3 Displaying personalisation at the layer and feature levels

Personalisation is provided at both the layer and feature level. Non-landmark layers are personalised at the layer level whereas landmark layers can be personalised at the layer and individual feature level. The following section displays maps that are personalised based on the profiles of users who have contrasting content preferences.

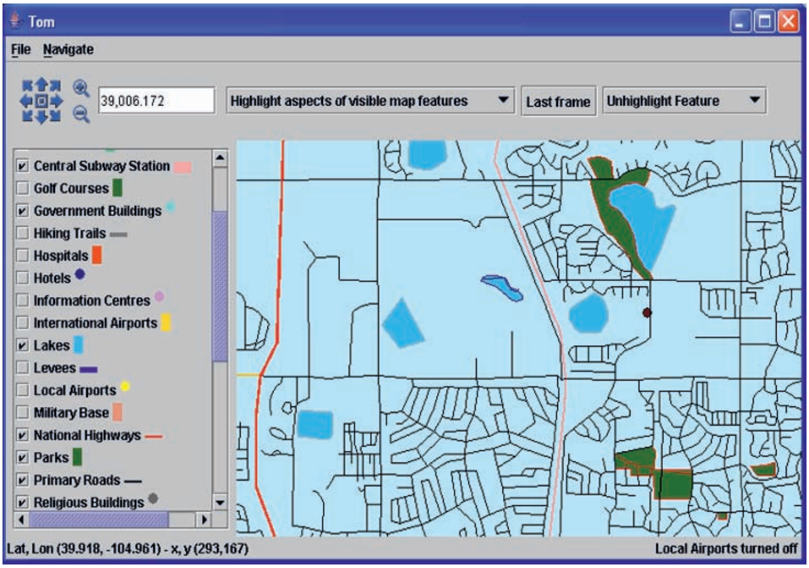


Fig. 7.5. Map showing layer level personalisation

Fig. 7.5 and 7.6 show maps that are personalised at the layer and feature levels respectively for a user with children and whose preferences centre on outdoor activities. As a result map layers like parks, lakes, and schools are recommended as map

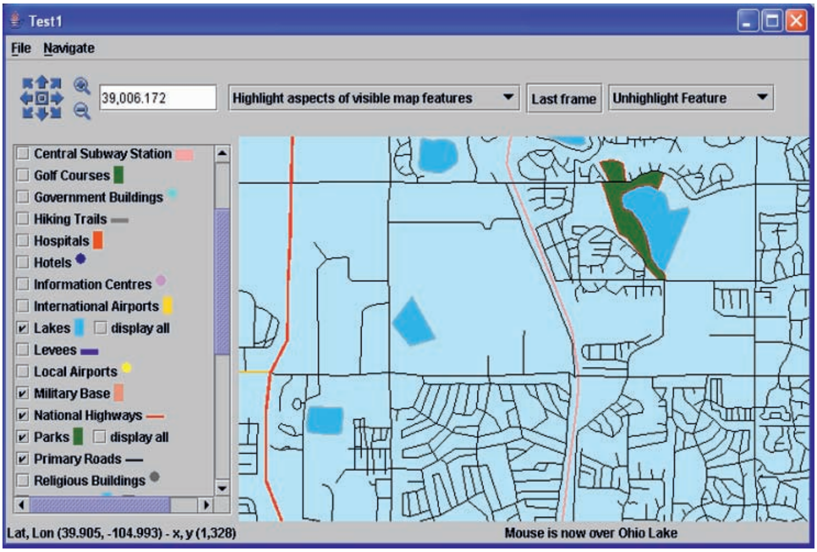


Fig. 7.6. Map showing feature level personalization

content of interest. Looking at Fig. 7.5 we can see a map region displaying all the parks, lakes, and schools for that region as the map has been personalised at the layer-level. In contrast, Fig. 7.6 displays a map of the same region showing the same landmark layers personalised at feature level δ_1. As can be seen from Fig. 7.6 there is a notable reduction in the number of schools, parks, and lakes present, as only those individual features with feature scores exceeding δ_1 are recommended.

Fig. 7.7. Personalised map with high personalisation threshold

Fig. 7.7 shows a map personalised at feature level δ_2 for a user whose profile describes them as a homemaker with children. δ_2 is set very high resulting in only the highest relevance features being returned to the user upon receiving a request for a map. As can be seen from the map the only landmark features present are apartment blocks (visiting friends), hospitals (taking kids to the doctor), shopping centres (shopping), and schools (dropping kids to school). It is possible to alter δ in order to display more or less detail depending on the preferences of the individual requesting the map.

7.5 Evaluating MAPPER efficiency

In previous experiments carried out (Weakliam et al, 2005b) it was shown that personalising map content at the layer level, in a manner similar to the personalisation technique described in this article, assisted the user when completing mapping tasks. Results of the experiments carried out in (Weakliam et al, 2005b) show that users were able to complete tasks with more ease when presented with personalised maps than when presented with non-personlised maps due largely to the recommendation of pertinent map layer content. It was also shown that the recommendations made by the application became more accurate as the number of mapping tasks completed by the participants increased. In conclusion prominent issues linked to both information overload and demands for explicit user input were effectively addressed during the experiment due to the efficiency of the personalisation provided.

An experiment was carried out to test the hypothesis that personalising maps at both the layer level and feature level benefits users when using MAPPER. Six participants took part in the experiment. Three of the participants had experience using the application, whereas the other three had not used the application on any previous occasions. The three participants who had no experience whatsoever interacting with the application were given a five-minute instruction on how to use the application. Each user was instructed to complete different mapping tasks over a period of one month where each task centered on specific map content. The users had complete freedom to interact with the maps presented to them, using any combination of map browsing actions, but ultimately had to complete the task assigned to them for that session. The maps returned were personalised using preference information extracted from user models, generated from user interaction history recorded from previous sessions.

The following displays results that show that personalising maps based on user interaction information captured implicitly can benefit users requesting mobile maps due to the considerable reduction in the size of datasets used to render the maps. Fig. 7.8 shows a chart of the various map types presented to the 6 experiment participants vs. the size of the dataset used to render the maps. In Fig. 7.8 *NP* represents a fully detailed non-personalised map (used as a control), *PL* represents maps personalised at the layer level based on preference information established from user interaction history, and *PF* represents maps personalised at both the layer and feature level based also on preference information determined from interaction history. In both *PL* and *PF* the number of recommended non-landmark map layers is set to 6, whereas the number of recommended landmark layers is set to 10. For *PF* δ is set to 0.25.

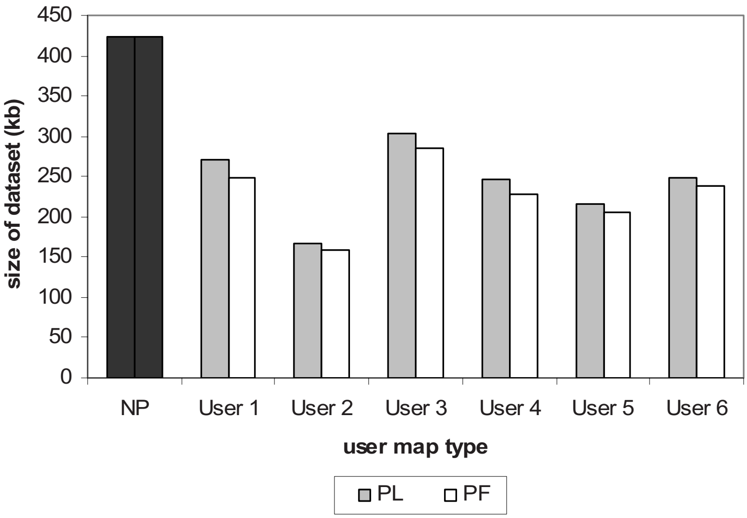


Fig. 7.8. Map type vs. size of dataset used to render map

Looking at Fig. 7.8 a significant decrease in the size of datasets used to render personalised maps both at the layer level and feature level is evident when compared to the non-personlised control. From examining results of the experiment described in (Weakliam et al, 2005b), it is important to note that the number of requests for additional layer content decreased as the number of tasks completed increased. This is primarily due to the fact that as the number of tasks completed increased, the system was able to ascertain user map content preferences more precisely as a result of continuous interaction between users and specific map layers. This has important consequences for the generation of personlised maps with MAPPER, as if a user is content with the level of detail presented to them, then the information recommended by the system is indeed accurate and is sufficient for the user to complete their task. This in turn addresses the problems of information overload and mobile device limitations.

7.6 Conclusions and future work

Humans encounter problems related to information overload and HCI when interacting with maps on mobile devices. When rendering maps on mobile devices developers are faced with several major difficulties, ranging from small screen sizes for map display to limited bandwidth for transmitting map data across wireless networks. In response to these problems we have designed and implemented MAPPER, which is a mobile application that generates personalised maps for users on the move at two distinct levels of detail. All map actions executed by users on the mobile device are captured implicitly and are used to infer user preferences in map feature content. User models are then created and updated dynamically based on user interactions with mobile maps. Personalising maps in this manner is extremely useful as it results in a considerable reduction in the size of datasets used to render maps on mobile devices. Reducing the size of map datasets allows the shortfalls of limited screen size, low computational power, and restricted bandwidth to be addressed and results in faster download times than if presenting users with fully detailed maps. This is paramount when users request maps when on the move.

For future work several key areas must be addressed. First of all we are transferring the full functionality of MAPPER to a more portable device than a Tablet PC, i.e. a PDA. We are also looking into improving the functionality available at the interface, e.g. implementing more complex spatial queries for professional users. Finally, more detailed user studies than those outlined in this chapter need to be carried out. This includes both qualitative and quantitative analyses of the system functionality. The impacts that further evaluations may have on MAPPER functionality must be assessed in order to improve MAPPER efficiency.

References

Agrawal, R., Imielinski, T., and Swami, A.N. (1993): Mining association rules between sets of items in large databases. *Proceedings of the ACM SIGMOD International Conference on Management of Data*, Washington, D.C, pp. 207-216.

Fink, J. and Kobsa, A. (2002): User modelling in personalised city tours. *Intelligence Review* 18(1), pp. 33-74.

Fischer, G. (2001): User modeling in human-computer interaction. *User Modeling and User-Adapted Interaction* 11(1-2), pp. 65-86.

Hinze, A. and Voisard A. (2003): Locations- and time-based information delivery in tourism. *Proceedings of the 8th International Symposium on Advances in Spatial and Temporal Databases*, Santorini Island, Greece, pp. 489-507.

Horvitz, E., Breese, J., Heckerman, D., Hovel, D., and Rommelse, K. (1998): The Lumiere Project: Bayesian user modelling for inferring the goals and needs of software users. *Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence*, Madison, Wisconsin, pp. 256-265.

Kelly, D. and Belkin, N. (2001): Reading time, scrolling and interaction: Exploring implicit sources of user preferences for relevance feedback during interactive information retrieval. *Proceedings of the 24th Annual International Conference on Research and Development in Information Retrieval* (SIGIR '01), New Orelans, LA, pp. 408-409.

Kelly, D. and Teevan, J. (2003): Implicit feedback for inferring user preference: A bibliography. *SIGIR Forum* 37(2), pp. 18-28.

Kim, J., Oard, D.W., and Romanik, K. (2001): User modelling for information access based on implicit feedback. *Proceedings of the ISKO France Workshop on Information Filtering*, Paris, France.

Linton, F., Joy, D., and Schaefer, H.P. (1999): Building user and expert models by long-term observation of application usage. *Proceedings of the International Conference on User Modelling* (UM99), Banff, Canada, pp. 129-138.

MapQuest (2006): http://www.mapquest.com/

OpenMap (2006): http://openmap.bbn.com/

Oppermann, R. and Specht, M. (2000): A context-sensitive nomadic information system as an exhibition guide. *Proceedings of the Second International Symposium on Handheld and Ubiquitous Computing* (HUC 2000), Bristol, UK, pp. 127-142.

Oracle Spatial: http://www.oracle.com/technology/software/products/spatial/index.html

Reichenbacher, T. (2001a): The world in your pocket – towards a mobile cartography. *Proceedings of the 20th International Cartographic Conference* (ICC 2001), Beijing, China, pp. 2514-2521.

Reichenbacher, T. (2001b): Adaptive concepts for a mobile cartography. *Supplement Journal of Geographical Sciences* 11, pp. 43-53.

Schmidt-Belz, B., Nick, A., Poslad, S., and Zipf, A. (2002): Personalised and location-based mobile tourism services. *Proceedings of Mobile HCI'02 with the Workshop on "Mobile Tourism Support Systems"*, Pisa, Italy.

Tiger/Line files (2006): http://www2.census.gov/geo/tiger/tiger2k/

Weakliam, J., Wilson, D., and Bertolotto, M. (2005a): Implicit interaction profiling for recommending spatial content. *Proceedings of the 14th International Symposium on Advances in Geographic Information Systems* (ACMGIS'05), Bremen, Germany, pp. 285-294.

Weakliam, J., Lynch, D.B., Doyle, J., Bertolotto, M., and Wilson, D. (2005b): Delivering personalized context-aware spatial information to mobile devices. *Proceedings of the 5th International Workshop on Web and Wireless Geographic Information Systems* (W2GIS'05), Lausanne, Switzerland, pp. 194-205.

Weisenberg, N., Voisard, A., and Gartmann, R. (2004): Using ontologies in personalised mobile applications. *Proceedings of the 12th annual ACM international workshop on Geographic Information Systems* (ACMGIS'04), Washington DC, pp. 2-11.

Yahoo! Maps (2006): http://maps.yahoo.com/

Zipf, A. (2002): User-adaptive maps for location-based services (LBS) for tourism. *Proceedings of the 9th International Conference for Information and Communication Technologies in Tourism* (ENTER 2002), Innsbruck, Austria, pp.329-338.

Zipf, A. and Richter, K.F. (2002): Using focus maps to ease map reading. Developing smart applications for mobile devices. *Künstliche Intelligenz* (KI). Special issue: Spatial Cognition (4), pp. 35-37.

8 A Survey of Multimodal Interfaces for Mobile Mapping Applications

Julie DOYLE[1], Michela BERTOLOTTO[1], David WILSON[2]

[1]School of Computer Science and Informatics, University College Dublin
[2]Department of Software and Information Systems, University of North Carolina at Charlotte

Abstract. The user interface is of critical importance in applications providing mapping services. It defines the visualisation and interaction modes for carrying out a variety of mapping tasks, and ease of use is essential to successful user adoption of a mapping application. This is redoubled in a mobile context, where mobile device limitations can hinder usability. In particular, interaction modes such as a pen/stylus are limited and can be quite difficult to use in a mobile context. Moreover, the majority of GIS interfaces are inherently complex and require significant user training, which can be a serious problem for novice users such as tourists. We propose an increased focus on developing multimodal interfaces for mobile GIS, allowing for two or more modes of input, as an attempt to address interaction complexity in the context of mobile mapping applications. Such interfaces allow users to choose the modes of interaction that are not only most intuitive to them, but also most suitable for their current task and environment. This chapter presents the user interaction problem and the utility of multimodal interfaces for mobile GIS. We describe our multimodal mobile GIS CoMPASS which helps to address the problem by permitting users to interact with spatial data using a combination of speech and gesture input. CoMPASS is set in the context of a representative survey across a range of comparable multimodal systems, and the effectiveness of our approach is evaluated in a user study which demonstrates that multimodal interfaces provide more intuitive and efficient interaction for mobile mapping applications.

8.1 Introduction

Intuitive Graphical User Interfaces are paramount when developing mobile applications providing map services. The availability and usage of mobile devices has increased dramatically in recent years and while mobile device technology has significantly improved since its beginning, there are still a number of limitations associated with such devices (e.g., small interface footprint, use in motion) which can hinder the usability of mobile mapping applications. Specifically, we are concerned with the limited interaction techniques mobile mapping users face, making it necessary to address human computer interaction challenges associated with mobile device technology when designing mobile geospatial interfaces. Indeed, restricted modes of interaction are a key factor of GIS interface complexity, which is another significant problem with current mobile mapping applications. This chapter advocates the design of multimodal interfaces for mobile mapping applications to address both the limited interaction techniques and interface complexity associated with such applications.

The benefits of multimodal interfaces, particularly within mobile geospatial environments, are numerous. Traditional input techniques such as a keyboard and mouse are unfeasible and inefficient in mobile environments. To counteract this problem mobile devices are equipped with a pen/stylus for interaction. However, the pen alone is not sufficient for expressive and efficient interaction. Mobile users are continuously in motion when carrying out field-based spatial tasks and their hands and eyes may be busy fulfilling such tasks. In such situations speech is an attractive input alternative to pen input, as it is more natural for users to speak and move than to point/input text and move. Multimodal interfaces allow users to choose the most appropriate modality for carrying out varied spatial tasks in contrasting environments. Users have the freedom to exercise control over how they interact. Therefore, they can choose to use the modality that not only is more suited to their current task, but also is most intuitive to them for this task. This has the benefit of greatly increasing the accessibility of multimodal applications to a wider range of users in various application contexts. For example, speech may not be the most ideal mode of input for an accented user or for a user with a cold. It can also be inappropriate or inefficient in certain environments such as a museum context or a noisy outdoor environment where the user is required to repeatedly issue commands until they are interpreted correctly. In these situations, using pen input/gestures may be more effective or acceptable. However, there are also limitations associated with pen input for users, for example, with repetitive strain injury or a broken arm. Moreover, users may find it difficult to point precisely to small interface objects or to select from interface menus, particularly on mobile devices such as PDAs. Therefore it is essential to design flexible multimodal interfaces that allow for two or more modes of interaction, for interactive mobile mapping applications.

Usability plays a vital role in a user's acceptance and adoption of a geospatial application. To ensure maximum usability, interfaces for such applications should be user friendly, intuitive to both novice and professional users alike and highly interactive. However, many GIS interfaces are intrinsically complex and require domain specific knowledge for carrying out map-based tasks. Research has shown that multimodal interfaces can aid in considerably reducing the complexity of GIS interfaces (Fuhrmann et al, 2005; Oviatt, 1996a). Multimodal interfaces have been an exciting research paradigm since Bolt's influential 'Put That There' demonstration (Bolt, 1980), which allowed for object manipulation through speech and manual pen input. Interest in multimodal interface design is motivated by the objective to support more efficient, transparent, flexible and expressive means of human-computer interaction. Multimodal interaction allows users to interact in a manner similar to what they are used to when interacting with humans. Using speech input, gesture input and head and eye tracking for example, allows for more natural interaction.

This chapter discusses issues that arise in the development of flexible, mobile mapping interfaces that allow mapping services to mobile geospatial users, providing multiple input modalities. The contribution of our chapter is two-fold. First, we describe CoMPASS (Combining Mobile Personalised Applications with Spatial Services), the mobile mapping system that we have developed on a Tablet PC. CoMPASS allows users to connect to a remote server and download vector maps in GML file format over wireless connections, to mobile devices. Users can then dynamically interact with these maps through pen and voice input modes. Available interactions include zooming and panning, querying and spatial annotating. Users can also manipulate the visual display by turning on/off features and changing the appearance of map features. Furthermore, they can attach annotations to spatial locations and features. CoMPASS relies on open-source libraries for GIS GUI development and therefore does not require the use of proprietary GIS software. In addition, speech recognition packages are widely available and can be easily integrated with existing code. The second part of this chapter is devoted to a representative survey of existing research systems describing multimodal mobile geospatial system development, including a comparison of the presented systems with CoMPASS.

The motivation behind our research is to overcome some of the challenges of mobile systems and issues of complexity of GIS interfaces. Allowing multiple input modalities addresses the problem of limited interaction capabilities and allows users to choose the mode of interaction that is most intuitive to them, hence increasing the user-friendliness of a mobile geospatial application.

The remainder of this chapter is structured as follows. Section 8.2 presents CoMPASS, the multimodal mobile mapping system that we are developing. The functionality of our application is described with close attention paid to the speech recognition module of our multimodal interface. In section 8.3 we provide a comprehensive overview of current state-of-the-art within multimodal interface design with particular focus on user evaluations of such systems. Details of a CoMPASS user study to determine the effectiveness and efficiency of our multimodal interface are presented in section 8.4, while section 8.5 outlines the results of this study. Finally section 8.6 concludes and discusses areas of possible future work.

8.2 The CoMPASS system

In this section we describe the multimodal GUI of the GIS prototype, CoMPASS (Combining Mobile Personalised Applications with Spatial Services) (Doyle, 2006a; Weakliam et al, 2005b), that we have developed on a Tablet PC. CoMPASS incorporates the delivery of vector map information using personalisation (Weakliam et al, 2005a) and Progressive Vector Transmission (PVT) (Bertolotto et al, 1999), as well as interactive augmented reality for media annotation and retrieval (Lynch et al, 2005), which is relevant to the user's immediate spatial location. The CoMPASS prototype has been fully implemented and tested. In this section we focus on the development of the Graphical User Interface of CoMPASS which allows for geospatial visualisation and interaction using multiple input modalities for mobile users. The following subsections detail the functionality of the CoMPASS multimodal interface. Emphasis will be placed on the speech module of CoMPASS which provides speaker independent speech recognition in real time.

8.2.1 Interacting with the data - CoMPASS multimodal interface

CoMPASS provides mobile mapping services for both novice and professional users within the field of GIS. Initial development was PDA-based but a number of factors, particularly the slow download and response time of maps, including personalised maps, caused us to concentrate our efforts on a Tablet PC implementation. A Tablet PC is a highly portable mobile device and provides superior viewing and editing capabilities for users in a mobile context. The Tablet PC we have used is a Compaq 1100 model with 1.1 GHz CPU chip, 512 MB DDR RAM, an 802.11b Wi-fi card supporting 11 Mbps and a 10.4 inch display. Our interface is based on that of OpenMap™ (OpenMap™, 2006). OpenMap™ is an open-source Java-based mapping toolkit that allows users to develop mapping applications in their own style.

A CoMPASS user logs onto the system via their username. Their current location is obtained via GPS and an interactive, personalised vector map is returned to them based on this geographical position. It is then possible for users to dynamically interact with their map. Available interactions include navigation through zooming and panning. This is possible through buttons on the interface but also by drawing an area on the map where the user would like to focus. It is also possible to re-centre the map at a particular location by simply clicking on that map location. Manipulation of map features is possible through turning on/off map features (such as highways, schools etc.) and changing the colour and size of map features for individual map feature content personalisation. CoMPASS also supports feature querying including highlighting specific features in an area defined by the user, highlighting a specific number of features closest to a given point, finding the distance between two points and multimedia annotation creation and retrieval.

Other aspects of system functionality include an unobtrusive information bar at the bottom of the interface displaying information on the user's current position (latitude and longitude) and the name of the feature the user is currently at (i.e. what feature the pen is over). This prevents text cluttering the interface which is of particular importance for mobile devices. Our user evaluation, described in section 8.4 demonstrates that this method of display is adequate even in a mobile context, as almost all users, though mobile during the evaluations, stopped walking while they were carrying out a specific task, allowing them to view the screen easily. CoMPASS provides a help menu to aid users in interacting with the system. All of the above-described functionality can be carried out using pen input, speech input or a combination of both. This ensures a flexible, easy to use interface as each individual user can choose the mode of input best suited to their expertise and context. Providing two or more modes of input with parallel functionality is of particular significance in mobile environments where a particular mode might be unsuitable or unfeasible. Fig. 8.1 and 8.2 depict user interactions with a map for both tourist and professional users respectively. The maps displayed are in vector data format. In Fig. 8.1, the user is a new user to the system and hence this map is non-personalised. The default scale for a non-personalised map is 1:8000. However, the user can adjust this scale to their preference. As the map is non-personalised, all possible map features are returned, which can give the appearance of a cluttered map. However, as the user interacts during future sessions over a period of time, CoMPASS learns their preferences and hence a user's map becomes personalised and contains less features (Weakliam et al, 2005a).

With regard to the design of the interface, as the system was initially PDA based it was decided to give the majority of the screen real estate over to map display. As a result, users would have a better view of their current and desired locations, allowing easier navigation between the two. However, subsequent development and evaluation of the system on a Tablet PC revealed that some users had difficulty pointing to and selecting small interface components (e.g. zooming and panning buttons) with the pen of the Tablet PC. For this reason, such interface components were enlarged, providing easier pen-based interaction. We believe that these changes address the usability concerns expressed by users when inputting via pen and, as such, should not have any biasing effect on the use of modalities in our system.

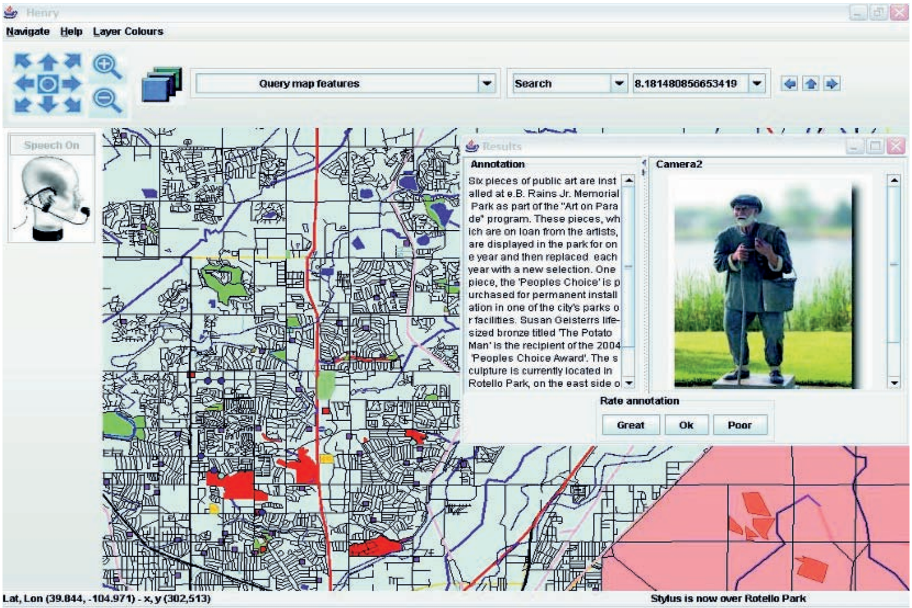


Fig. 8.1. Screenshot of tourist viewing an annotation regarding an art exhibition in a local park

8.2.2 The speech and gesture module

We have integrated a speaker-independent speech module into the CoMPASS system, which is capable of human-computer interaction handling in real time. This module depends on the use of a commercially available speech recognition package, IBM's ViaVoice™ (*IBM,* 2006). If the user wishes to interact using speech input, they must specifically turn the speech recognition engine on by clicking the "Speech On" button located on the interface. An icon then appears (Fig. 8.1) indicating to the user that it is now possible to interact via speech. CoMPASS responds by delivering an audio message to the user, informing them that they can issue the command "help" to view a list of available commands for interacting (Fig. 8.3). Two modes of speech input are available when interacting with the CoMPASS interface – voice commands and dictation.

Voice commands consist of short phrases made up of one or two words. We felt that keeping voice commands short would reduce the time to learn these commands and hence increase the efficiency of the system, as users would not be reliant on the help menu. Such phrases are matched against a specified rule grammar, which contains a list of all possible voice commands available for interacting with the system. Providing a specific set of voice commands ensures more precise and robust recognition, as an interface action will only be carried out if the command associated with the action has been recognised and determined as being a legitimate voice command.

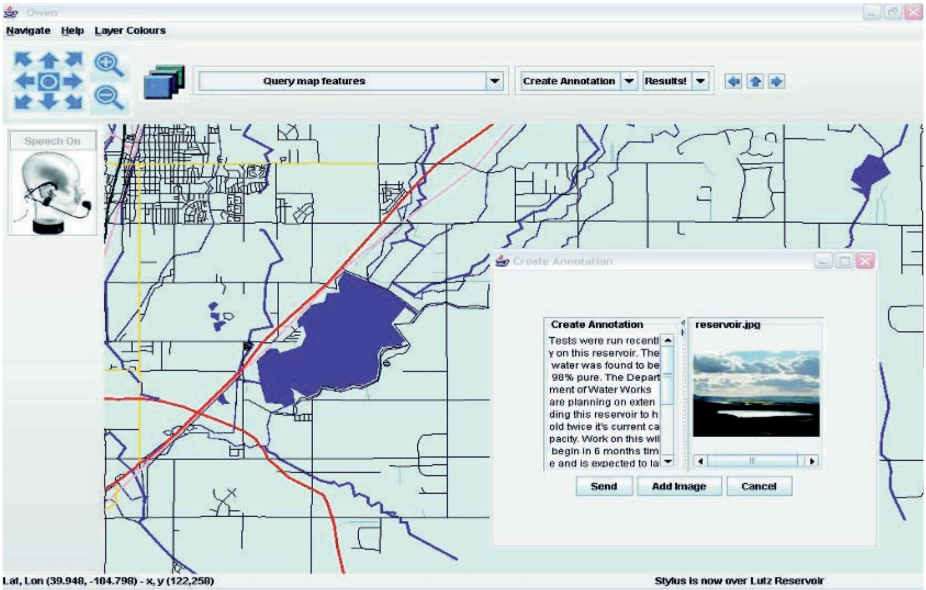


Fig. 8.2. Screenshot of a surveyor creating an annotation (using pen and keyboard) regarding a particular reservoir

Voice Commands

Currently there are approximately 350 commands that CoMPASS recognises. The vast majority of these commands contain a feature name, combined with another word for performing some action on that feature. For example, voice commands can be used within CoMPASS for navigating (e.g. "zoom in", "pan northwest"), feature manipulation (e.g. "parks red", "highways on") and querying spatial features (e.g. "highlight lakes", "find distance"). One aspect of our system functionality that should be highlighted is that the user receives visual feedback after they have issued a command. The command is displayed on the left hand side of the information bar allowing the user to check that their command has been interpreted correctly. Similarly, once the required action has been carried out on the interface, a message is displayed on the information bar to notify the user of such (Fig. 8.3). Providing some form of feedback to users plays a crucial role in assuring them that their intentions have been interpreted correctly and that the task they were hoping to achieve has been completed successfully. This in turn enhances system usability and intuitiveness. Querying requires a combination of sequential voice commands, pen gestures and speech feedback. For example, if a user issues the command "Find distance" to find the distance between two distinct points on the map CoMPASS responds by asking the user to 'Please draw a straight line on the map'. It was decided to use such a combination of speech and pen for queries as research has shown that while speech is useful for issuing commands and describing features it is not so intuitive for describing spatial locations and objects (Oviatt, 2003). Pen gestures are generally better suited to such tasks. However, of interest for further development of the speech recognition component would be to search for a particular place name on the map, for example 'Take me to Rotello Park'. There is currently no mechanism within our application to search for place names; a user simply must navigate through the map and point to features on it to discover that feature's name. A searching mechanism could considerably increase the efficiency of a user's task. Combined speech and pen gestures are also used for annotating features on a map. This is described below.

Dictation

The CoMPASS speech module can process dictation entered by a user. Dictation is essentially free-form speech and so enforces fewer restrictions on the user regarding what they can say. Such input is not matched against a rule grammar, but rather a dictation grammar. Dictation is used within CoMPASS for annotating map features. Once a user issues the command "Annotate", the rule grammar is disabled and the dictation grammar becomes active. CoMPASS responds by delivering an audio message informing the user that they should input their voice annotation. Once the user has finished speaking, their voice annotation is displayed on the information bar, providing feedback as to whether or not each word of their annotation was correctly recognised (Fig. 8.4). The system delivers a second audio message asking the user to confirm that their annotation is correct. If the user provides confirmation, they are requested to pick a point on the map to assign the annotation, whereupon the annotation and its spatial location are recorded by the system.

However, as dictation grammars contain no specific rules pertaining to what the user can say, they tend to be more error prone. It is likely, particularly in outdoor mobile environments, that one or more of the words spoken during dictation will not be recognized correctly. For example, in Fig. 8.4 the user entered the voice annotation "Rotello Park has an excellent art exhibition". However, the system interpreted this as "*Retail* Park has an excellent *card* exhibition". Hence, it becomes crucial to provide methods for the user to correct their voice annotation if necessary. It has been recognised (Suhm et al, 2001; Oviatt, 2000a) that allowing the user to switch modality during continuous speech error correction can result in increased correction accuracy. This process is referred to as "multimodal error correction". CoMPASS leverages this technique in its multimodal interface. The system requests spoken confirmation from the user regarding the correctness of their dictated annotation. If the user indicates that the annotation is erroneous, the system responds by advising the user that they can now correct any errors. A window containing the possible modes of error correction is displayed and the user must choose from re-speaking, using the pen and virtual keyboard of the Tablet PC or handwriting (Fig. 8.4). Each of these modes allows the user to correct the individual words that have been imperfectly recognized.

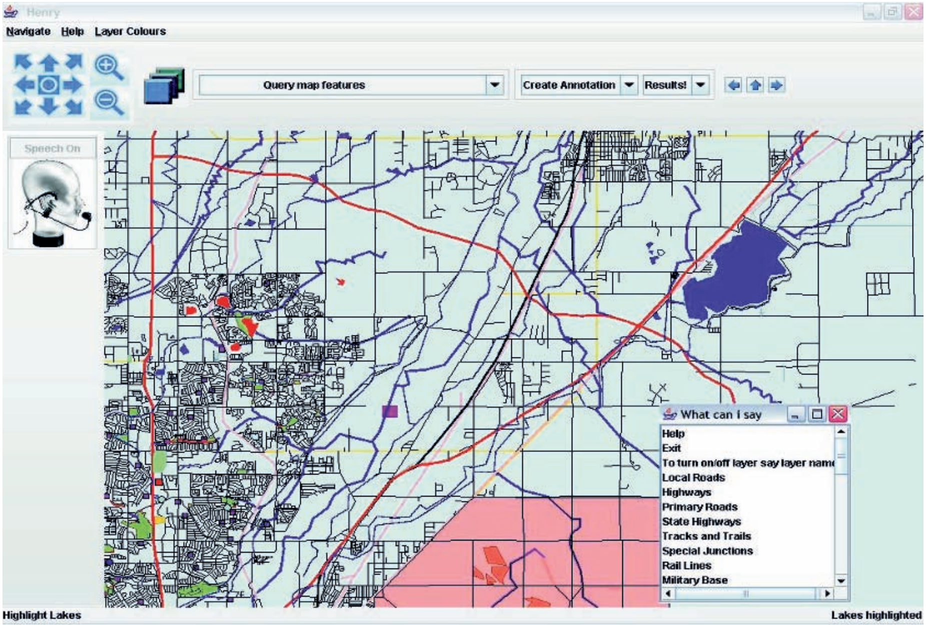


Fig. 8.3. This screenshot displays the result of the 'highlight lakes' command. Once the action has been carried out, the user is informed through a message printed to the information bar

Gesture and Handwriting

In addition to voice commands and dictation, CoMPASS also recognises and processes gestures and handwriting. Gestures can take the form of ‘intra-gestures’ i.e. pointing or selecting with the stylus to locations or objects on the Tablet PC screen. ‘Extra-gestures’ that allow users to point to surrounding objects in their current environment are not supported. Intra-gestures can take two forms within CoMPASS: pointing and dragging. Users can point at objects to re-centre the map at this point, to discover the name and type of objects, to specify what feature they would like to query or what feature they would like to annotate. Dragging gestures specify a ‘zoom in’ on the area over which the pen is dragged or, when used in conjunction with a query, to specify the area of interest for the query. Handwriting can be used within CoMPASS as a method to correct errors during dictation of annotations. The handwriting recogniser can process both block and cursive handwriting. If a word is not recognised correctly, the user can choose from a list of alternatives simply by clicking on the word. We have yet to evaluate the efficiency of and preference for handwriting as a mode for error correction. Should it prove favourable with users, handwriting might be considered as an alternative mode of initial input for annotations, rather than simply a correction mode.

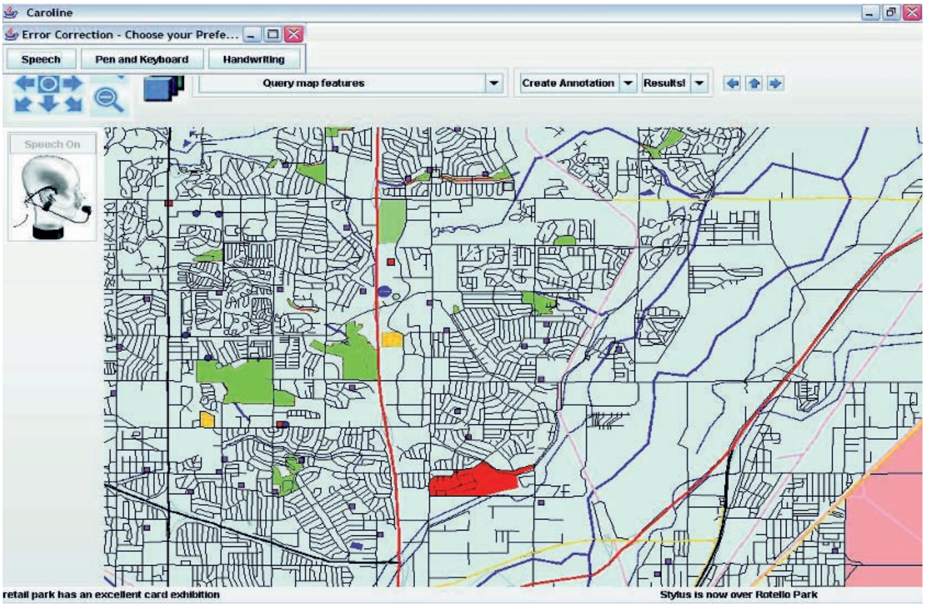


Fig. 8.4. Here the user has entered an annotation using dictation. The recognized annotation has been printed to the information bar. In this case the annotation is incorrect and the user informs the system which responds by displaying the error correction window

8.3 Survey of existing methodologies

Multimodal interfaces are a new class of interfaces that aim to identify naturally occurring forms of human language and behaviour, and which integrate one or more recognition-based technologies such as speech, pen and vision (Oviatt, 2003). Such interfaces process two or more combined user input modes in a coordinated manner with multimedia system output. Significant advances have been made in developing multimodal interfaces in recent years. This has been due, in large part, to the multitude of technologies available for processing various input modes and to advances in device technology and recognition software. A varied set of multimodal applications now exist that recognize various combinations of input modalities such as speech and pen (Oviatt, 2003), speech and lip movements (Benoit et al, 2000), and vision-based modalities including gaze (Qvarfordt et al, 2005), head and body movement (Nickel et al, 2003) and facial features (Constantini et al, 2005).

In addition, the array of available multimodal applications providing map services has broadened widely and ranges from city navigation and way-finding for tourists, to emergency planning and military simulation. This section focuses on providing a representative survey of the state of the art within the research realm of multimodal interfaces, for applications providing mobile mapping services. In particular we analyse the methodologies used for evaluating such interfaces. We will focus our attention on mobile multimodal systems that process active forms of input i.e. speech and pen.

Such methods of interaction are more reliable regarding user intention than vision-based (passive) methods, which require no input from the user but instead implicitly monitor their movements to infer recognition (Oviatt, 2003).

8.3.1 Multimodal tour guide applications

A mobile architecture for pedestrian navigation and designed to support several modalities for presentation and user input is outlined in (Wasinger et al, 2003a) and (Wasinger et al, 2003b). This system incorporates the fusion of different modes of input such as speech, gesture and keyboard/mouse as well as sensors. The system allows users to utilize all modes simultaneously. The aim of such a system is to increase speech recognition rates in environments where noise levels are high and user contexts are constantly changing. As a result, the system incorporates both static and dynamic rule grammars for speech recognition. Dynamic grammars are those that model landmark and street objects visible on the map, allowing for interaction with such objects via the object's name. In addition, a speech synthesizer provides audio route instructions to users in navigating to a desired destination. Initial results indicate media fusion produced consistently more robust recognition rates than when inputs were used separately (Wasinger et al, 2003b). However, apart from navigation and viewing information on landmark objects, there exists very little interactivity between the user and their map. The system presented by Wasinger et al is that of a tour guide, which provides navigation support for pedestrian tourists. CoMPASS on the other hand provides tourist and professional users alike with a complete GIS. That is, multimodal functionality is supported that provides users with the capability to not only navigate, but also query, manipulate and annotate their physical location.

Further research from Wasinger and colleagues describes a personal navigation system that can span different situations and a mobile Shopping Assistant (Kreuger et al, 2004), (Wasinger and Kreuger, 2004). An integrated navigation service, the BMW Personal Navigator (BPN) allows users to plan a trip from their home PC, transfer the downloaded information to a PDA which can then connect to an in-car navigation system. At this point, the car's navigation system takes over and guides the user to their required destination. When on foot, the PDA plays a much more important role to guide the user to their final destination, whether outdoor or indoor. The user can receive both graphical and verbal route instructions through the mobile user interface. Moreover, multimodal interaction is supported on the PDA, providing users with the ability to navigate and interact with objects in their environment through speech and gestures (both intra and extra), although full GIS functionality is not supported. One of the underlying services of the BPN system is the indoor Mobile ShopAssist (MSA) to aid users in choosing and buying a digital camera. In addition to speech and gesture input, the MSA also recognises handwriting input.

The SmartKom system was developed by a German research consortium encompassing partners from both research and industry. Their aim was to develop intuitive methods of interaction to support several input modalities including speech, gesture and facial expressions. There are three application scenarios of SmartKom (1) SmartKom Home/Office (2) SmartKom Public and (3) SmartKom Mobile (Malaka et al, 2004), a PDA-based application for navigation and information for tourists. In this chapter we focus on the third solution. A user evaluation study of SmartKom Mobile is presented in (Jost et al, 2005), the results of which aim to answer the questions: (1) What are the most suitable interaction techniques for navigational and informative tasks for mobile pedestrians and (2) Do social and situational context affect multimodal interaction? The user study was carried out both in laboratory conditions and in an outdoor context. A total of 20 subjects participated.

Each subject was asked to carry out three tasks for the evaluation comprising of map navigation, planning a tour and locating information about sights. Additional GIS functionality such as querying, manipulating and annotating was not considered. Users trained themselves on the system by performing the above tasks using both speech only input and a combination of speech and gesture. During the training task, users were given a list of possible speech commands for interacting with the system. In the evaluation phase, subjects were requested to use multimodal interaction to perform tasks, but were not provided with a list of available speech commands. This is in contrast to CoMPASS where users can issue the "help" voice command or open a help menu item to view a list of all possible commands for interacting with the system. This helps to increase task efficiency, as users do not have to 'guess' what command they can use. Each SmartKom Mobile evaluation subject completed a short questionnaire after each of the 3 individual tasks covering interaction details, followed by a final questionnaire dealing with various aspects of the system's functionality and use.

In the questionnaires, users were asked to compare 5 usability parameters of interacting with the system to the best other method they knew for performing a similar task. These parameters were: (1) convenience of completing the task; (2) speed of completing the task; (3) intuitive usability of the multimodal interaction; (4) efficiency of the interaction and (5) overall acceptance of the system. Overall, the results showed that users fundamentally accept multimodal interaction in a mobile information system. The majority of users reported increased convenience, speed and usability while interacting multimodally. The only exception to this was for users performing a map interaction task in laboratory conditions. One possible explanation provided by the authors for this exception is that users miss traditional GUI components more in laboratory settings than in the field, as most people are used to interacting with traditional GUIs while indoors. With regard to multimodal interaction in particular social or location contexts, users reported they would rather interact multimodally when other people, particularly strangers, are not around. This finding is consistent with the CoMPASS user evaluations.

8.3.2 Evaluations of multimodal systems

One of the earliest systems providing a multimodal interface to a mobile GIS was QuickSet developed by Cohen et al (Cohen et al, 1997), the first prototype of which was designed and built in 1994. QuickSet is a collaborative, multimodal (pen/voice) interface for map-based tasks with application to distributed systems. The QuickSet system allows users to create entities on a map by simultaneously speaking and drawing – the user speaks the name of the entity to be created and indicates its location on the map using a pen. For example, to create a unit on the map the user would hold the pen at the preferred location and say: “red T72 platoon”. This would result in the creation of a new unit of the specified type on the map interface. The user can annotate the map and create points, lines and polygons of various types using speech input, pen input or a combination of both. Speech and pen input can be recognised in parallel. Partial information from these two input streams is then fused and the system searches for the best joint interpretation of the multimodal input. In this way, individual input modes can disambiguate (Oviatt, 1999) one another aiding in suppression of errors. QuickSet’s multimodal interface also supports confirmation allowing the system to correct errors of recognition or interpretation.

QuickSet provides a multimodal interface to a number of distributed applications including military simulation, virtual reality and medical informatics, on a handheld PC. The architecture of QuickSet is agent-based and the application uses a set of collaborating agents including (1) a continuous speaker-independent speech recogniser (recognition of complex noun-phrases and a small number of sentence forms is possible); (2) a gesture recogniser; (3) a multimodal natural language processor; (4) a unification based multimodal integration component and (5) a map-based interface. This type of architecture ensures the system is scalable to various hardware configurations.

A user study to compare the efficiency of using a traditional direct-manipulation GUI with that of QuickSet is discussed in (Cohen et al*,* 2000). Four subjects, retired US military domain experts, participated. Following a brief training session each subject was asked to complete a task on both the traditional interface, using keyboard and mouse, and the QuickSet interface, employing speech and pen input. The mean time required for task completion on each interface was calculated as were the total number of errors, error correction rates and the recognition rate of QuickSet. Results showed considerable efficiency advantages (a 3.7-fold increase) for multimodal input over traditional input methods when interacting with GUIs. Moreover, error correction was four times more costly with a traditional GUI. Another advantage of QuickSet was that users reported a strong preference for multimodal interaction. However, the QuickSet evaluation differs from that of CoMPASS in that our evaluation was field-based. CoMPASS users evaluated the system whilst mobile, in both outdoor and indoor environments, providing results regarding application use in mobile contexts.

Relevant work on the development and evaluation of multimodal user interfaces was initiated by Sharon Oviatt over a decade ago. Oviatt has conducted many significant user studies demonstrating the efficiency of and user preference for multimodal interaction during map-based tasks. One of Oviatt’s earliest studies (Oviatt, 1996a) aimed to evaluate human performance in terms of efficiency, accuracy, cognitive load and preference when interacting with an application providing a map-based service, namely a Service Transaction System simulating real estate tasks. Eighteen subjects participated in the user study and completed 2 subtasks within each of the following conditions: (1) speech-only communication; (2) pen-only communication and (3) multimodal speech/pen communication. For each task, Oviatt measured the total number of both spoken and written words, the number of disfluencies, the number of sentences containing spatial location descriptions, the task completion time, the total number of task-critical errors and self-reported and observed preferences. The evaluation was carried out on an LCD tablet. However unlike subjects during the CoMPASS evaluations, Oviatt’s subjects were not mobile whilst interacting.

As a result of the user study presented in (Oviatt, 1996a), Oviatt concluded that multimodal systems can "permit sufficient flexibility for users to avoid many errors and to optimise speed through selection of a particular input mode or integrated use of modes at appropriate points during an interaction". Compared to speech-only interaction, it was found that combined speech and pen input resulted in the elimination of 36% of content errors and 50% of spoken disfluencies. Moreover, task completion time was 10% faster during multimodal interaction. Oviatt attributes the poor performance, in terms of error content and efficiency, of speech-only input to subjects' difficulty in articulating spatially oriented descriptions. Indeed 48% of all content errors were related to incorrectly defining/locating spatial objects on the map. During multimodal interaction however, users avoided such errors by using the pen to point to/select spatial objects, thus eliminating the need to express such descriptions verbally. Of great significance was the discovery that 95% of users preferred interacting multimodally as opposed to unimodally. 5% of subjects indicated a preference for pen-only interaction but no users reported a preference for speech-only interaction. All CoMPASS subjects (i.e. 100% of users) stated they preferred multimodal interaction to pen-only input.

In a later study (Oviatt, 2000b) Oviatt undertook a user evaluation to ascertain (1) whether mutual disambiguation is beneficial in a mobile environment and also (2) whether the rate of mutual disambiguation is higher during mobile than stationary use. As explained in section 8.1, mutual disambiguation of two input signals allows each input mode to provide partial information that aids in the interpretation of the other mode. Sixteen subjects participated in the user study which involved completing tasks using the QuickSet multimodal system. The evaluation was carried out in two contrasting environments, a quiet room with little variation and a moderately noisy cafeteria environment where the user was continuously mobile and experienced variable lighting conditions and occasional interruptions. Users' multimodal commands were scored for a number of dependent measures including the rate of mutual disambiguation per subject, the multimodal recognition rate in both conditions and the total percentage of multimodal commands for which both speech input and gesture input was correct. It was established that 50-100% higher levels of mutual disambiguation were achieved in noisy mobile environments compared to quiet stationary environments. Furthermore, results showed that in a mobile environment, error rate was reduced by 19-35% using a multimodal architecture as opposed to using speech recognition as a stand-alone application. As was expected, speech recognition rates dropped 5% during mobile tasks, compared to stationary tasks. There was no significant difference in gesture recognition rate between the two conditions, while multimodal recognition rates degraded by 3% when users were mobile. These results show that multimodal systems can be developed that support significant levels of mutual disambiguation in mobile usage contexts.

The research provided in (Rugelbak et al, 2003) describes an experiment carried out to evaluate a multimodal interface designed for a handheld tourist guide. The intent of such research is to obtain knowledge about user behaviour with an application that supports simultaneous, coordinated speech and pen input. This type of multimodal interaction allows users to speak and tap simultaneously and these actions are then interpreted together. The participants for the study presented in (Rugelbak et al, 2003) consisted of seven user interface experts as it was thought such subjects could provide constructive criticism as to how the interface could be improved prior to user testing with novices. Each user was asked to carry out a series of tasks using speech and pen separately, as well as simultaneously. However, only four of the seven subjects used the two modalities simultaneously. Results of the user study showed that it was not obvious or intuitive that the interface was multimodal or that the two modalities could be used simultaneously. In contrast, the multimodal interface of CoMPASS has been shown to be both user-friendly and intuitive to non-expert users. CoMPASS users are always aware of when they can interact via speech, as they must specifically request to turn voice recognition on.

8.4 CoMPASS evaluation

In order to evaluate CoMPASS we undertook a user study to ascertain the efficiency of interaction using a combination of input modalities (speech and pen), compared to that of interacting using just pen input. The study was carried out in two different settings, the first being a quiet environment. For this study, an indoor setting was used. The second setting was a noisy environment, where we repeated the study in both a canteen and an outside area. We chose these areas for two reasons; firstly because of restrictions on the range of the wireless network in our university and secondly because both areas provided high noise levels. In all cases, participants were mobile i.e. walking around whilst interacting. Subjects were equipped with a wirelessly connected Tablet PC and a headset with a microphone for speech input, though the headphones of the headset were not used during the evaluation.

8.4.1 Subjects

A total of 12 volunteers participated in our user studies, 6 of these in our quiet environment study and 6 in the noisy environment study. While the number of available subjects is smaller than what we would have hoped to find (50 subjects were asked to participate), for an initial evaluation of our system we felt this number would give a good indication of whether our system was well received and whether the functionality provided allowed users to carry out their tasks efficiently and effectively. A follow-up evaluation with more users tested this further (Doyle, 2007a), (Doyle, 2007b). The subjects were students from a computer science research institute. The evaluation design was crossed factorial with one within-subject factor: the environmental context (i.e. noisy vs. quiet) and two between subject factors: Gender (male vs. female) and Accent (foreign vs. native English). Table 1 provides more detailed subject statistics.

8.4.2 User tasks

Each individual participant was assigned a specific task. An example of such a task is: “You are a tourist and you want to see some interesting buildings before going shopping. Plan a route between some government buildings and find the nearest shopping centre to the last building you visit.” Each task was then subdivided into a list of actions to be carried out to achieve the task (Tables 8.2 and 8.3). While the tasks carried out by each user were distinct, the number of subtasks performed was the same. Each test subject was asked to interact twice with the system to carry out his or her required task: during the first interaction participants interacted unimodally (using the pen as the sole mode of input), while during the second interaction participants used a combination of both speech and pen input. While the tasks were the same for both conditions, the user’s current position (on the map) was changed for the second interaction, requiring the user to navigate to find the points of interest. As such, we perceived no accumulative effect of the first interaction on the second one. During both interactions, subjects performed the same number of actions so as to ascertain an accurate recording of which mode of interaction was most efficient.

Table 8.1. Subject Statistics

	Quiet Environment	**Noisy Environment**
Number of Males	3	3
Number of Females	3	3
Average Age	26.8	24.5
Non-Native Speaker	1	0

Table 8.2. Unimodal Interaction with a Map (Pen)

	Action/Task Performed
1.	Zoom and Pan (using pen)
2.	Turn off 2 non-relevant features
3.	Query the map – highlight a feature or find the distance between two features
4.	Change the colour of a feature type
5.	Create a textual annotation and assign it to a feature on the map

Table 8.3. Multimodal Interaction with a Map (Speech and Pen)

	Action/Task Performed
1.	Zoom and Pan (using pen)
2.	Turn off 2 non-relevant features using voice commands
3.	Query the map – highlight a feature or find the distance between two features – using a combination of voice commands and pointing/selecting gestures
4.	Change the colour of a feature type using voice commands
5.	Create an annotation using voice dictation and pointing gestures to assign the annotation to a feature on the map

As the majority of subjects were not familiar with the application or Tablet PC architecture, a short 10-minute training session was provided to demonstrate the functionalities and interaction techniques possible within the system. Each user was then timed and monitored as they carried our both their tasks. In addition to task completion time, some other data was recorded including:

- The number of system errors during both tasks.
- The number of user errors during both tasks.
- The average number of voice commands spoken during the second task.
- The average number of times it was necessary to repeat commands and
- The average number of mis-recognitions.

Finally, all participants completed a questionnaire covering aspects of their interaction during each task, ease of use of the system and graphical user interface.

8.5 Results

Our main interest for this research was to ascertain if using multimodal interaction provided an increase in efficiency while interacting with a mobile GIS. We hypothesised that interaction using a combination of speech and pen input as opposed to solely pen input, would improve task efficiency across all experimental factors. We were also interested in the user's subjective experience of system usability, intuitiveness of the multimodal interaction and their preference for uni or multimodal interaction. With regard to preference, we expected our results to confirm those of Oviatt discussed in section 8.3.2 (Oviatt, 1996a), i.e. that the majority of users would state a preference for multimodal as opposed to unimodal interaction. However, what sets our evaluation and results apart from those systems described in our survey is that CoMPASS provides multimodal functionality for a complete range of GIS functionality. As such, in contrast to SmartKom and the tourist guide applications presented in section 8.3.1, CoMPASS users were required to not only navigate and view information regarding objects in their vicinity, but to also query, manipulate and annotate map features in their physical space. Moreover, our results are based on users being mobile whilst carrying out their tasks, differentiating our evaluation from that of QuickSet (Cohen et al, 2000) and Oviatt (Oviatt, 1996a).

8.5.1 Interaction speeds

When testing interaction speeds we wanted not only to examine the overall difference between unimodal and multimodal interaction, but also to establish the effects of gender and environment on multimodal interaction rates. These results are presented in Fig. 8.5.

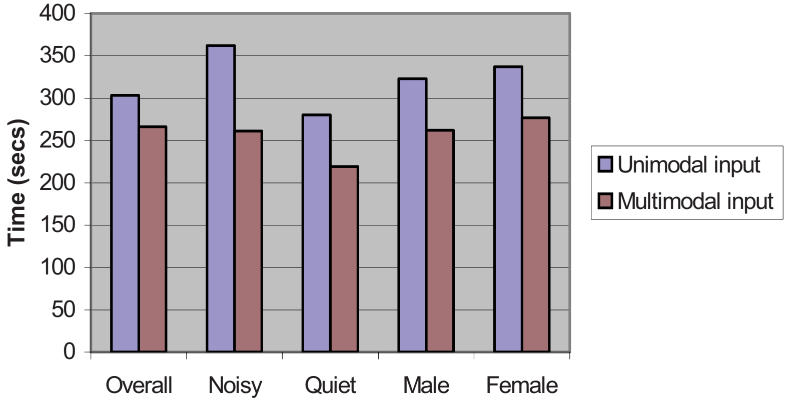


Fig. 8.5. Average interaction speeds for unimodal and multimodal interaction

These results show that across all factors, multimodal interaction using speech and pen was significantly faster ($p < 0.01$), and therefore more efficient than using the pen as a single modality. Overall, there was an increase of 12.21% when subjects used a combination of speech and pen input. This increase in efficiency whilst interacting multimodally is made even more significant as none of the subjects had previous experience using the multimodal interface of the system. Therefore, they were not familiar with the voice commands required to perform certain actions. We can consequently expect that as users become more familiar with the system efficiency rates will increase. There was no significant difference in multimodal interaction speeds between males and females. Efficiency increased by 18.89% and 17.8% respectively, for both groups of subjects interacting multimodally as opposed to unimodally. As expected, there was quite a significant increase in multimodal interaction speed for users interacting in a quiet, as opposed to a noisy, environment. As can be seen from Fig. 8.2, the average time taken to complete tasks multimodally in a quiet environment was 219 seconds, compared to 261 seconds in a noisy environment, an increase of 16.09%. This increase in speed can be attributed to the fact that for subjects in a quiet environment it was only necessary, on average, to issue voice commands once, whereas in a noisy environment subjects repeated themselves 2 to 3 times. Subjects were also instructed, whilst interacting multimodally, to perform an action using the pen if the system did not recognize their voice command after 3 attempts. This also affected interaction speeds in noisy environments. One surprising result is the large difference (101 seconds) in interaction speeds of unimodal (pen) input compared to multimodal input, in a noisy environment. It is unclear why this difference is so notable. One possible explanation is that while speech recognition clearly degrades slightly in noisy environments (as can be seen from our results), the use of a rule grammar allowing only specific voice commands means that any voice input picked up, other than that of the user, will not be recognized and so no action will be carried out on the interface. Thus the user does not have to spend time correcting errors. In addition, it is possible that concentration might decline in a noisy environment, explaining why unimodal interaction speed in this context is significantly higher than in a quiet context. Further testing is necessary to determine an actual cause. The standard deviation of speeds was small across all factors, as all subjects completed the same number of tasks and actions.

8.5.2 Error rates

Also of interest to us in our evaluation was comparing the error rate for unimodal interaction against that of multimodal interaction. Overall, error rate was considerably low, indicating that our multimodal interface was easy to learn and intuitive to use. Across all tasks, there were no system errors whilst users interacted using solely pen input. A total of 3 system errors, across two subjects, were recorded for multimodal tasks. These errors corresponded to mis-recognitions, for example the system interpreted the command "Lakes Red" as "Lakes Off" and so turned off lakes rather than changing their colour to red. Only one user-error was recorded during all multimodal tasks, which consisted of the user issuing an incorrect command to the system (i.e. a command that was not in the rule grammar). However four user errors were recorded for unimodal interactions. Two of these were related to querying using a pen while another two were related to navigating (i.e. zooming out instead of zooming in and panning in the wrong direction due to imprecise pointing to the correct GUI component).

8.5.3 Users' experiences

After the conclusion of their tasks, each subject completed a questionnaire on various aspects of the study so as to allow the evaluators to gain an insight into each participant's subjective experience whilst interacting. Most importantly, we were interested in whether users preferred interacting using one mode of input, i.e. the pen, or interacting multimodally. We found that all subjects preferred the latter. Reported reasons for this preference included feelings of increased control over how one could interact, the fact that the pen was better suited for certain tasks (like selecting/pointing), and speech was more suited to others and that voice commands were brief and easier to use than the pen. Many participants stated that problems with interacting with the pen related mainly to scrolling lists and pointing precisely to some GUI components such as the zoom and pan buttons. To this effect we have redesigned the interface, enlarging those components that were deemed to have caused problems during pen interaction. However, we expect that these problems regarding pen usability may have had an impact on our interpretation of our results. A follow-up evaluation will investigate this further. 75% of subjects found interacting with the map unimodally more time consuming than multimodally while 75% to 80% stated that voice commands were more natural than pen input. With regard to ease of use of the system, 62.5% felt that task 1 (pen-only interaction) was easier to complete than task 2. The main reason given for this was that users simply were not used to interacting with a system using voice commands. Of these 62.5%, 55% agreed that with increased usage, interacting multimodally would become easier.

8.6 Discussion

Within the research realm of mobile interactive mapping applications, multimodal interfaces are a relatively new and exciting concept. Such interfaces significantly assist in increasing the flexibility and robustness of mobile GIS and in addition have been shown to enhance the usability of such applications for a broad range of users. This chapter presented a detailed overview of various multimodal applications providing mapping services, together with our own application, CoMPASS. In contrast to the array of applications discussed in this chapter, CoMPASS leverages full advantage of the multimodal paradigm for GIS by supporting a wide range of geospatial functionality. Furthermore, in addition to supporting multimodal input, CoMPASS also provides multimedia feedback to users as well as providing methods for multimodal error correction. Results of our user study indicate that such a comprehensive multimodal interface plays a vital role in the intuitiveness, user-friendliness and efficiency of our mobile mapping application. Hence, we expect that as mobile mapping applications become more widely available in the near future, an application such as CoMPASS will play a vital role in improving interaction with and usability of such systems.

While the benefits of a multimodal system are many, there are a number of factors an interface designer must keep in mind when developing a multimodal interface. Most importantly, multimodal interfaces can be disadvantageous on small mobile devices where providing and supporting multiple modes of input can increase the demands of a mobile application considerably. As such, multimodal applications may be more beneficial on a Tablet PC device, which has superior processing capabilities compared to a PDA or mobile phone. Moreover, ambiguity in noisy environments can be introduced rather than resolved through the use of more than one input mode. It must also be noted that although research shows that the majority of users do prefer to interact multimodally, some may not and interfaces should be designed to reflect this. For example, some users may not like or feel comfortable interacting through speech in public places and as such may not require the speech recognition engine to run in the background. Similarly, there are certain environments where speech is not appropriate, for example for tourists visiting churches or museums. Providing interface components that allow a user to explicitly turn speech recognition on/off can combat this problem. Similarly, interface components should be large enough to allow for easy interaction with a pen/stylus should a user wish to interact via pen input alone. Evaluations concentrating on solely pen interaction with such components would allow to determine their ideal size.

While the CoMPASS GUI has been fully implemented, we are currently looking into how we can enhance functionality, including methods of multimodal error correction within our system. Research (Suhm et al, 2001) has shown that unimodal repair is less accurate than multimodal error correction and for dictation tasks, such as creating annotations; multimodal correction is more efficient than unimodal by respeaking. An evaluation of this aspect of our system will determine if the entire process of inputting annotations via voice, including any necessary error correction, is more efficient than inputting annotations using solely pen input while in the field. In addition, a larger-scale user evaluation has just been completed, the aim of which was to both further evaluate the multimodal aspect of our system, and to examine certain interface issues such as the pen usability problem. Also of interest is synchronous rather than sequential fusion of inputs, as if a user currently attempts to speak at the same time as gesturing, errors will occur. Implementation of the system on additional mobile platforms is also ongoing for broader evaluation.

References

Benoit, C., Martin, J.C., Pelachaud, C., Schomaker, L. and Suhm, B. (2000): Audio-visual and Multimodal Speech-based Systems. In *Handbook of Multimodal and Spoken Dialogue Systems: Resources, Terminology and Product Evaluation,* D. Gibbon, I. Mertins and R. Moore (Eds.), pp. 102-203, Kluwer.

Bertolotto, M. and Egenhofer, M. (1999): Progressive Vector Transmission. In *7th ACM Symposium on Advances in Geographic Information Systems,* pp. 152-157, Kansas City, Kansas.

Bolt, R. A. (1980): Put-That-There: Voice and Gesture at the Graphics Interface. In *The 7th International Conference on Computer Graphics and Interactive Techniques,* pp. 262-270, Seattle, Washington.

Cohen, P.R., Johnston, M., McGee, D., Oviatt, S., Pittman, J., Smith, I., Chen, L. and Clow, J. (1997): QuickSet: Multimodal Interaction for Distributed Applications. In *5th ACM Conference on Multimedia,* pp. 31-40, New York, New York.

Cohen, P., McGee, D. and Clow, J. (2000): The Efficiency of Multimodal Interaction for a Map-based Task. In *6th International Conference on Applied Natural Language Processing,* pp 331-338, Seattle, Washington.

Constantini, E., Pianesi, F. and Prete, M. (2005): Recognising Emotions in Human and Synthetic Faces: The Role of the Upper and Lower Parts of the Face. In *10th International Conference on Intelligent User Interfaces,* pp. 20-27, San Diego, California.

Doyle, J., Weakliam, J., Bertolotto, M. and Wilson, D. (2006a): A Multimodal Interface for Personalising Spatial Data in Mobile GIS. In *ICEIS – 8th International Conference on Enterprise Information Systems,* in press, Paphos, Cyprus.

Doyle, J. (2007a): Multimodal Interface Design for Mobile GIS. A PhD thesis submitted to University College Dublin, January 2007.

Doyle, J., Bertolotto, M. and Wilson, D. (2007b): Multimodal Interface Design for Geospatial LBS. Under review for the Journal of Location Based Services.

Fuhrmann, S., MacEachren, A., Dou, J., Wang, K. and Cox, A. (2005): Gesture and Speech-based Maps to Support Use of GIS for Crisis Management: A User Study. In *AutoCarto 2005,* Las Vegas, Nevada.

IBM ViaVoice™ (2006): http://www-306.ibm.com/software/info1/websphere/index.jsp?tab=products/mobilespeech

Jost, M., Haussler, J., Merdes, M. and Malaka, R. (2005): Multimodal Interaction for Pedestrians: An Evaluation Study. In *10th International Conference on Intelligent User Interfaces,* pp. 59-66, San Diego, California.

Krueger, A., Butz, A., Muller, C., Stahl, C., Wasinger, R., Steinberg, K.E. and Dirschl, A. (2004): The Connected User Interface: Realizing a Personal Situated Navigation Service. In *The 9th International Conference on Intelligent User Interfaces (IUI 04),* pp. 161-168, Madeira, Portugal, ACM press.

Lynch, D., Bertolotto, M. and Wilson, D. (2005): Spatial Annotations in a Mapping Environment. In *GISRUK '05, GIS Research UK,* pp. 524-528, Glasgow, Scotland.

Malaka, M., Haeussler, J. and Hidir, A. (2004): SmartKom Mobile – Intelligent Ubiquitous User Interaction. In *9th International Conference on Intelligent User Interfaces,* pp. 310-312, Madeira, Portugal.

Nickel, K. and Stiefelhagen, R. (2003): Pointing Gesture Recognition based on 3D-tracking of Face, Hands and Head Orientation. In *5th International Conference on Multimodal Interfaces,* pp. 140-146, Vancouver, Canada.

OpenMap™ (2006): http://openmap.bbn.com/

Oviatt, S. (1996a): Multimodal Interfaces for Dynamic Interactive Maps. In *SIGCHI Conference on Human Factors in Computing Systems,* pp. 95-102, Vancouver, Canada.

Oviatt, S. (1999): Mutual Disambiguation of Recognition Errors in a Multimodal Architecture. In *The Conference on Human Factors in Computing Systems (CHI '99),* pp. 576-583, Pittsburgh, Pennsylvania.

Oviatt, S. (2000a): Taming Recognition Errors with a Multimodal Interface. In *Communications of the ACM,* Vol. 43, No. 9, pp.45-51.

Oviatt, S. (2000b): Multimodal System Processing in Mobile Environments. In *13th Annual ACM Symposium on User Interface Software and Technology,* pp. 21-30, San Diego, California.

Oviatt, S. (2003): Multimodal Interfaces. In *Handbook of Human-computer Interaction*, J. Jacko and A. Sears (Eds.), pp. 286-304, New Jersey.

Qvarfordt, P. (2003): Spoken Feedback in Multimodal Interaction: Effects on Users Experience of Qualities of Interaction. In *1st Nordic Symposium on Multimodal Communication,* pp. 21-34.

Qvarfordt, P. and Zhai, S. (2005): Conversing with the User based on Eye-Gaze Patterns. In *SIGCHI Conference on Human Factors in Computing Systems,* pp. 221-230, Portland, Oregon.

Rauschert, I., Agrawal, P., Sharma, R., Fuhrmann, S., Brewer, I., MacEachren, A., Wang, H. and Cai, G. (2002a): Designing a Human-Centered, Multimodal GIS Interface to Support Emergency Management. In *10th ACM International Symposium on Advances in Geographic Information Systems,* pp. 119-124, Virginia.

Rauschert, I., Sharma, R., Fuhrmann, S., Brewer, I. And MacEachren, A. (2002b): Approaching a New Multimodal Interface. In *2nd International Conference on GIS (GIScience),* Colorado.

Rugelbak, J. and Hamnes, K. (2003): Multimodal Interaction – Will Users Tap and Speak Simultaneously? In *Telektronikk '03,* pp. 118-124.

Schapira, E. and Sharma, R. (2001): Experimental Evaluation of Vision and Speech based Multimodal Interfaces. In *Workshop on Perceptive User Interfaces,* pp. 1-9, Florida.

Suhm, B., Myers, B. and Waibel, A. (2001): Multimodal Error Correction for Speech User Interfaces. In *ACM Transactions on Computer-Human Interaction,* Vol. 8, No. 1, pp. 60-98.

Wasinger, R. and Kreuger, A. (2004): Multimodal Interaction with Mobile Navigation Systems. In *Information Technology,* Vol. 46, No. 6, pp. 322-331, Oldenbourg Publishers.

Wasinger, R., Stahl, C. and Kreuger, A. (2003a): M3I in a Pedestrian Navigation and Exploration System. In *The 4th International Symposium on Human Computer Interaction with Mobile Devices (Mobile HCI 03)*, pp. 481-485, Udine, Italyl, Springer-Verlag LNCS.

Wasinger, R., Stahl, C. and Kreuger, A. (2003b): Robust Speech Interaction through the use of Multiple and Different Media Input Types. In *Eurospeech 2003,* pp. 1049-1052.

Weakliam, J., Bertolotto, M. and Wilson, D. (2005a): Implicit Interaction Profiling for Recommending Spatial Content. In *ACMGIS '05,* pp. 285-294, Bremen, Germany.

Weakliam, J., Lynch, D., Doyle, J., Bertolotto, M and Wilson, D. (2005b): Delivering Personalized Context-Aware Spatial Information to Mobile Devices. In *W2GIS, 5th International Workshop on Web and Wireless Geographic Information Systems,* pp. 194-205, Lausanne, Switzerland.

9 User Interaction in Mobile Navigation Applications

Kristiina JOKINEN

Department of Computer Sciences, University of Tampere

Abstract. The chapter focuses on cooperation and interaction in multimodal route navigation and provides an overview of the advantages and disadvantages of multimodal interaction in location-based services in general. The goal of the research has been to study methods and techniques for richer human-computer interaction, and to investigate interconnection and user preferences concerning speech and tactile input modalities on a route navigation task. The chapter also surveys the work on a mobile navigation application which allows the user to query public transportation routes using speech and pen pointing gestures. The first version of the PDA-based navigation application, MUMS, has been developed with the Helsinki City public transportation as the domain, and the user can ask timetable and navigation information either by natural language questions or clicking on the map. On the basis of user studies, we also discuss the individual modalities and their influence in interactive applications.

9.1 Introduction

In this chapter we discuss the design of multimodal and interactive route navigation systems from the point of view of conversational cooperation. Multimodality is seen as a natural extension of speech applications in order to make them more flexible, and it is also crucial in the development of location-based services, such as route navigation, which allow the user to get information related to their location. The departure point is natural interaction, which does not only refer to natural language interaction, but also to those aspects of interaction that are realised in the presentation of information to the user and providing the user with enhanced possibilities for task management and cooperative communication.

The first sections of the chapter discuss multimodality and its advantages in practical interactive systems. Section 9.2 introduces the concept of cooperation in human-human and human-computer interactions, and sets the wider context for the rest of the paper. Section 9.3 discusses multimodality and defines terminology, while section 9.4 provides necessary background concerning multimodal human-computer interaction and multimodal systems in general. We then move to multimodal interaction in mobile route navigation systems. Section 9.5 discusses various aspects of map navigation and route presentation in connection with the requirements for location-based service systems. We discuss aspects concerning different wayfinding strategies, cognitive load, and technical properties. Special attention is paid to issues concerning multimodality in mobile context. As a specific example of multimodal map navigation systems, section 9.6 presents an overview of the multimodal route navigation system MUMS, developed in the Finnish national technology project (New

Methods and Applications for Speech Technology). MUMS has been developed for a PDA, and it uses the Helsinki city public transportation system as its domain. Main results of the evaluation of the MUMS system are also repored, with the main questions concerning the user's preference of one input mode over the other, and the system's usefulness in terms of task completion and user satisfaction. Another important question deals with the user's perception of the control of the system which is assumed to relate to the user's impression of the cooperative nature of the system. Finally, section 9.7 draws conclusions of the application development and finishes with future prospects and ponderings concerning further research directions to develop interactive mobile navigation applications.

9.2 Cooperation and grounding

It may seem somewhat far-fetched to consider principles of human-human communication as the starting point of interactive system design. However, the fact that most practical speech-based applications suffer from inflexibility and lack of user-friendliness mean that there is need for better understanding of what is meant by naturalness in interactive situations. Speech also adds an extra demand on the interaction since it produces an illusion of natural language communication: the users intuitively exploit human communication strategies and can often see real value in the use of spoken language as an interface (instead of typical direct manipulation interfaces) only if the system would perform more fluently. Moreover, in recent years a new metaphor has risen to describe interaction with computers: that of the computer as an agent which assists human users in dealing with complicated systems and large amounts of information (see discussion e.g. in Jokinen, 2004). It can thus be expected that by studying how such a fundamental aspect of human-human communication as cooperation could be modelled in the applications, it would be possible to make interaction less frustrating and intuitively more satisfying.

What kind of cooperation are we talking about? In dialogue modelling research, intentional collaboration has been much studied in the frameworks of Teamwork (Cohen and Levesque, 1991) and SharedPlans (Grosz and Sidner, 1990). While dialogue interactions, especially task-based service dialogues, also include collaboration, some authors, e.g. Allwood (1976), distinguish it from cooperation. Cooperation is a special attitude of the speakers, stemming from their rational agenthood: it is part of fundamental dialogue behaviour, and based on rational coordinated action in general. For instance, Grice (19xx) formulates four conversational maxims which characterise the Cooperative Principle, while Jokinen (1994), following Allwood, talks about Ideal Cooperation as the starting point for successful communication. In relation with interactive services, we follow the latter line of thinking, and regard cooperation as a sign of naturalness of the interface, distinguishing it from the usability of the interface which refers to the general fitness for the task. Cooperation can be seen as emerging from the system's processing capabilities which maintain interaction with the user on the basis of relevant and truthful information. It can also be extended to cover those properties of the system that enable the user to interact with the system in an appropriate way such as the physical interface (mouse and keyboard, mobile phone, PDA) and robust processing and presentation of the task-related data (Jokinen, 2004).

Consider the following human-human dialogue between a Helsinki City Transportation service agent and a customer. The dialogue is a translated, slightly simplified telephone dialogue from the Interact corpus (Jokinen et al. 2002).

Dialogue (1)

A: I'd like to ask about bus connection to Malmi hospital from Herttoniemi metro station – so is there any possibility there to get a bus?
L: Well, there's no direct connection – there's the number 79 that goes to Malmi but it doesn't go to the hospital, it goes to Malmi station
A: Malmi station? oh yes – we've tried that last time and it was awfully difficult
L: well, how about taking the metro and changing at Sörnäinen, or Hakaniemi if that's a more familiar place
A: Well Hakaniemi is more familiar yes
L: Ok, from there you can take the bus 73
A: 73?
L: Yes it leaves Hakaniemi just there where you exit from the metro to the bus stops, next to the market place
A: So it's by the market place that 73 leaves from?
L: Yes
A: And it's not there where the other buses leave from in front of Metallitalo?
L: No, it's there right when you come out from the metro
A: And it goes to the hospital?
L: yes, it has a stop just by the hospital
A: Ok, it must be a better alternative than the bus we took to the station, we didn't know which way to continue and nobody knew anything and we had to take the taxi…
L: what a pity – there would have been the number 69 though. It leaves close to the terminal stop of number 79 and goes to the Malmi hospital.
A: I see, so 79 to the station and then 69?
L: yes
A: Are they on the same stop?
L: well not on the same stop but very close to each other anyway
A: close to each other? Ok, well thank you for your help.
L: thank you, goodbye
A: goodbye

Besides the apparent vagueness and spoken language syntax, Dialogue 1 features non-deterministic dialogue structure and mixed-initiative dialogue strategies. The agent L's flexible guidance for the best route is an example of considerate interaction strategies. Since the customer A had found the route via Malmi station "awful", L first introduces the simplest route via Hakaniemi and guides the customer through the route using metro and bus. Once the customer has returned to her earlier frustrating experience of going to the Malmi station by bus, L also provides information of this option which requires going by two buses and some walking. It is this kind of sensitiveness to the partner's needs and attending their emotional stress that characterizes cooperation in human communication. The agent L does not impose her superior knowledge of route options nor submit herself to the partner's bad choices, but smoothly provides information of the best alternatives and calmly sorts out the bad alternative without insisting on the same type of information.

The dialogue also shows another type of cooperative communication related to the context in which the dialogue takes place. The speakers make frequent references to the physical environment (*change in X*, *close to each other*, *Hakaniemi*, *Malmi station*), and the spatial and visual environment directs their interpretation and generation of the language. In other words, language is *grounded* in the communicative context. Grounding is so natural to human communication that we do not pay attention to the frequent situational and contextual references in the language use, nor the whole range of different modalities used in processing and manipulating information in our every-day interactions (gestures and pointing, nodding, gazing, etc.). Grounding is regarded as one of the most important problems in AI (see Harnard (1990) for a discussion of how a symbolic system can learn the connections between symbols like words and the real world entities like the referents of the words), and e.g. the VITRA project (Maaß, 1993, 1995), studied the grounding of language into the environment, especially how visual and spatial information is referred to in natural language utterances.

In dialogue research, the term grounding is also used in relation to cooperation and collaboration by referring to the speakers' establishment of the common ground (Clark and Schaeffer, 1989). The speakers constantly monitor their contributions and provide the partner with implicit or explicit feedback concerning understanding and acceptance of the issues being talked about. In Dialogue (1), grounding is exemplified by several confirmation questions and short acknowledgements (*ok*).

In the design of practical speech-based systems, the computer is usually regarded as a tool which supports human goals: it is passive and it should function in a transparent way under human control. Dialogue (2) shows a typical interaction with a speech-based service system (modified from an interaction with a real timetable application). Compared with Dialogue (1), interaction strategies in Dialogue (2) are designed to compensate for less than perfect speech recognition (see e.g. Danieli and Gerbino 1995, Walker et al. 2000), and also to make sure that the users respond by utterances which are relevant and recognisable (see Yankelovich, 1996). Grounding is modelled explicitly by giving feedback (*ok*), and implicitly by repeating task-relevant parameters (departure and arrival places).

Dialogue (2)

S: Welcome to the bus timetable and route information service.
Where would you like to leave from?
U: From Railway Station
S: Ok. Where would you like to go to from the Railway Station?
U: Alppila.
S: Ok. You want to go from Railway Station to Alppila.
The bus number 17 leaves at 14:38.
The tram number 3B leaves at 14:40.
The bus number 23 leaves at 14:56.
Would you like to get more information?
U: Yes.
S: Ok. What would you like to know? Say "next" if you want to hear timetables for the next buses, or "repeat" if you want to hear the information again, or "navigate bus number" if you want to hear route information about a particular bus number, or "new" if you want to have a new search, or "help" if you want to get more assistance.
U: New.
S: Ok. New search. Where would you like to leave from?

U: From Alppila
S: Ok. Where would you like to go to from Alppila?
U: Railway Station
S: Ok. You want to go from Alppila to Railway Station … …

The necessary context is usually tacitly included by the designers in the general usage scenario, but the system can also explicitly ask the user for relevant contextual information. Both techniques can lead to the user's frustration, however: the assumed context may not be the one the user is in, or the system repeatedly asks for information that seems obvious for the user in the given situation (cf. the end of Dialogue 2). Moreover, although it is possible to build spoken dialogue systems that conduct rather natural sounding conversations with the user, references to the physical environment especially in route navigation tasks can become rather clumsy, if they must be expressed purely verbally (cf. the long descriptions in the human telephone Dialogue 1: *there where you exit from the metro to the bus stops next to the market place*; *it's not there where the other buses leave from in front of Metallitalo*). On the other hand, it's also been shown that people construct referring expressions collaboratively: while the first reference is usually a long description, consecutive references tend to become shorter as the speakers share the context and common language to refer to the entities (Clark and Wilkes-Gibbs, 1986; Heeman and Hirst, 1995).

Multimodality offers a natural way to expand spoken dialogue systems' interaction capabilities and provide the users with more flexible and enjoyable interfaces. Especially for location-based services and tasks where spatial information is referred to, such as map navigation and way-finding, multimodal interfaces form a natural alternative. A map that both the user and the system share and can manipulate (within the limits of the task) presents a natural way of addressing the language grounding problem and of completing the underlying task in an efficient manner. Spoken language interaction provides fast and efficient input communication while pointing on the map helps the user to overcome some problematic situation, most notably errors in name recognition and missing user expressions. If both the user and the system can use input and output modalities that fit with the task, this should contribute to more flexible and enjoyable communication.

From the technical point of view, location-aware systems can also be seen as responding to the grounding problem, in that they enable the environment and the user's location be taken into account in the interaction design. Although it may still be a long way to go to engineer interactive location-based services with robust dialogue capabilities, there is active research in the location-based systems (cf. this book), and various scenarios with existing services have been studied (e.g. the Finnish NAVI-program, see Ikonen et al., 2002, Kaasinen et al., 2001). After defining the concept of multimodality and discussing multimodal systems and the specific characteristics of map-based navigation, we will come back to future prospects of mobile systems and location-based services.

9.3 What is multimodality?

Even though multimodality sounds like a straightforward concept, it is good to start with terminological clarification. In navigation systems research, multimodality usually refers to multiple modes of transport, i.e. different routes between two points using car, train, or walking. In this chapter, we consider multimodality in terms of modes of interaction. The concept of modality is understood as a particular channel through which communication is conveyed. However, a closer look reveals that the term is used to refer to slightly different things depending on whether we look at it from the human-centred or the computer-centred angle, and consequently, multimodality, or use of multiple modalities, can denote different things in interaction design.

In human communication, the focus is on perception and control, and the term *modality* is used to refer to the various sensory systems through which incoming data is transformed to higher-level representations so that the manipulation of environment can take place. We thus have the visual, auditory, tactile, olfactory, gustatory, and vestibular modalities, associated with appropriate sensory organs. We can also make finer distinctions along the modalities and talk about neurobiological modalities such as somatic senses and muscle sense (Shepherd, 1988), and it can also be argued that people think differently about written language than they do about diagrams (Stenning and Oberlander, 1995), and thus the two actually function as two different modalities even though both use the visual channel. There is symmetry between input and output modalities in the human-centred perspective, although it is more common to talk about human output in terms of gestures, facial expressions, speech, etc., instead of visual, auditory, etc. output, i.e. we think about the encoding and presentation of information rather than the modalities through which it will be perceived by partner.

Also in human-computer interaction, modality can be understood as referring to the computer subsystems that correspond to the use of human sensory channels, with the various input/output devices analogous to sensory organs. However, it is also useful to make a distinction between the devices through which information is received and transmitted (camera, graphical displays, loudspeakers, vibrating mouse, etc.) and the encoding of data in different media using different coding systems. For instance, computer displays can show text, graphics, and video, all of which are transmitted through visual modality, but use different coding systems to present the information.

In general, we can talk about multimodal systems as system which accept many different inputs and combine them in a meaningful way, i.e. they can use multiple input devices (multi-sensor interaction) or multiple interpretations of input issued through a single device, and also present information through different output devices or through different codings on a single device. Nigay and Coutaz (1995) and Vernier and Nigay (2000) emphasize the system-oriented aspects of their definitions: modality refers to the type of communication channel used to convey information. They distinguish physical device (loudspeaker, screen) and interaction language (natural language, graphical animation), and can thus extend the range of multimodal systems: a system is multimodal if it employs several interaction languages even though it uses only a single device. Furthermore, they require that a multimodal system can extract and convey meaning automatically, thus making a difference between multimodal and multimedia systems: the former strives for meaning, while the latter only uses various media channels to present information.

The ISLE/NIMM standardization group for natural multimodal interaction (Dybkjaer et al. 2002) also assumes a computer-oriented position but uses a two-way definition by conflating the code and modality. According to them, medium is the physical channel for information encoding such as sounds, movements, etc., while modality is a particular way of encoding information in some medium. For them, text, graphics, and video would all be different modalities on computer screen, and spoken language a special type of modality encoded in audio media.

We consider it important to distinguish code (interaction language) from modality, and also be consistent with the human-oriented comprehension of modalities so that the term refers to different types of sensory information. We thus follow Maybury and Wahlster (1998) who offer the following definitions:

- Medium = material on which or through which information is captured, conveyed or interacted with (i.e., text, audio, video)
- Code = system of symbols used for communication (language, gestures)
- Mode, modality = human perceptual systems that enable sensing (vision, auditory, tactile, olfaction, taste).

Graphics displayed on the computer screen is thus an instance of graphical output medium perceived through visual modality, while speech uses audio medium (microphone, loudspeakers) and auditory modality. Their definition of modality has also been criticized, since it does not readily correspond to the way the term has been used in the literature on multimodal systems. In the strictest sense, a system would need to process input that comes through two senses in order to be regarded as multimodal, and thus e.g. pen-based systems that use only pen would not be multimodal even though the input can be graphics and language, since both of these are perceived visually. However, the notion of code distinguishes these cases: drawings and textual words apparently follow different symbolic interpretations, and, following the extended definition of a multimodal system by Nigay and Coutaz, such a system could be called multimodal as it employs several output codes (interaction languages).

Fig. 9.1 clarifies multimodal human-computer interaction and corresponding modalities for the human users and the computer system (cf. Gibbon et al. 2000). The horizontal line divides the figure along the human-computer interface and shows how different modalities correspond to different input/output media (organs, devices) both on the human side and on the computer side. The figure can also be divided vertically so as to present the symmetrical situation between human cognitive processing and automatic processing by the computer. The input sides correspond to perception of the environment and analysis of sensory information into representations that form the basis for cognitive and information processing. The output sides correspond to the coordination and control of the environment through signals and actions which are reactions to input information after data manipulation. The figure shows how input and output channels correspond to each other when looking at the human output and computer input side (automatic recognition) and computer output and human input (presentation) side.

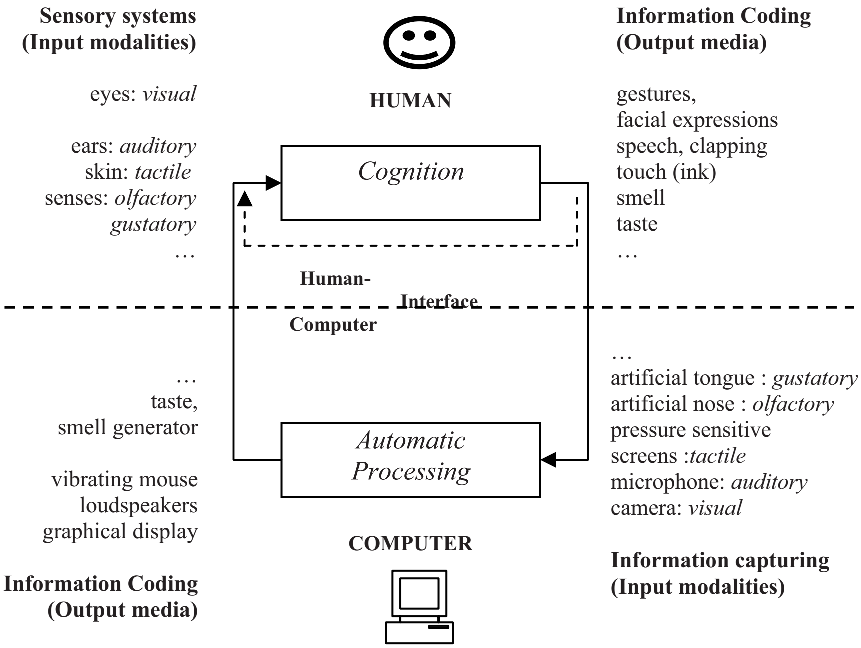


Fig. 9.1. Human-computer interface and different input/output modalities. The arrows represent information flow and the dotted arrow the human intrinsic feedback loop. Modified from Gibbon et al. (2000).

One final word needs to be said about natural language. Language is regarded as a particular form of symbolic communication, i.e. a system of linguistic signs. It may be often useful to consider natural language as a special type of modality, but we will go on with the terminology just introduced: natural languages use sound or movements (gestures) as media, they are transmitted and perceived through auditory or visual modalities, and they encode messages in specific natural language codes such as Finnish, English, or sign language.

9.4 Multimodality in human-computer interaction

9.4.1 Multimodal system architectures

Multimodal systems allow the users to interact with an application using more than one mode of interaction. Following Nigay and Coutaz (1995), the EAGLES expert and advisory group (Gibbon et al., 2000) defines multimodal systems as systems that represent and manipulate information from different human communication channels at multiple levels of abstraction. They distinguish multimodal systems from multimedia systems which also offer more than one device for the user input to the system and for the system to give feedback to the user (e.g. microphone, speaker, keyboard, mouse, touch screen, camera), but do not process the information on abstract representation levels.

Fig. 9.2 shows a detailed example of the conceptual design of multimodal system architecture (Maybury and Wahlster 1998). The system includes components for processing information on several abstraction levels as well as taking care of the component interaction and information coordination.

The upper part of the figure shows analysis components: the media input processing, media/mode analyses, and multimodal fusion. These components take care of the signal-level processing of input via different input devices, integration and disambiguation of the imprecise and incomplete input information, and interpretation, respectively. The lower part presents planning components which include multimodal design, media and mode synthesis, and media rendering through output devices. Multimodal design is an important part in deciding the general characteristics of the presentation style, and it includes content selection as well as media design and allocation. The actual realisation as cohesive output is taken care of by the synthesis components, such as natural language generator, speech synthesizer, and character animator.

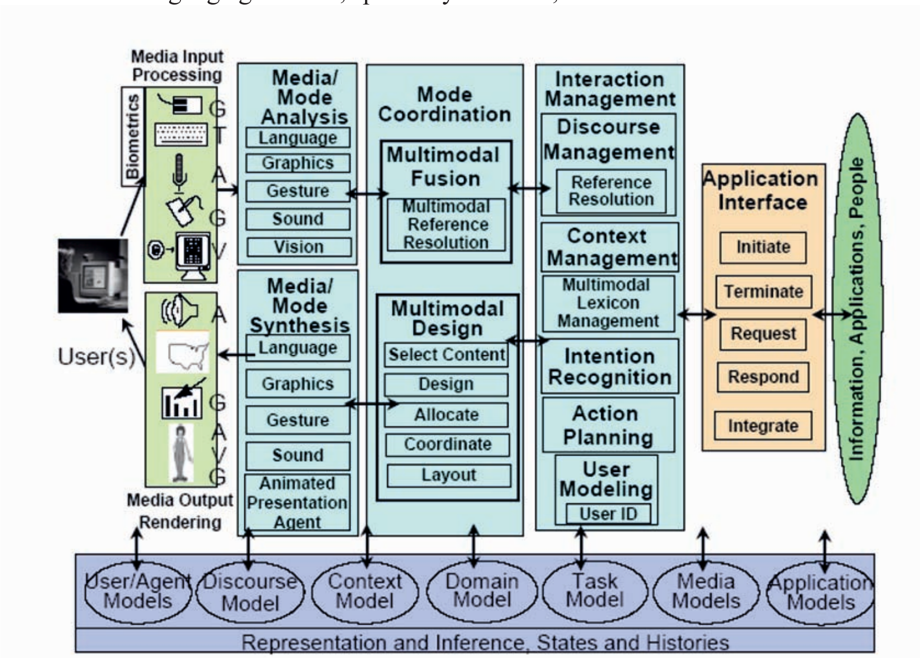


Fig. 9.2. An example of the multimodal system architecture. Developed at the Dagstuhl Seminar *Coordination and Fusion in Multimodal Interaction*, 2001, the original is based on Maybury and Wahlster (1998).

Interaction management deals with the characteristics of user interaction and application interface. The right side of the figure shows these components: discourse and context management, intention recognition, action planning, and user modelling. They all require knowledge of the dialogue context and presuppose particular methods and techniques that allow the system to reason on the basis of its knowledge. Discourse management deals with issues such as reference resolution and tracking the focus of attention, while context management includes managing both the spatial context of the user (possible map interface and local environment) and the temporal context of the interaction (dialogue history, the user's personal history, world knowledge). Intention recognition and action planning concern high-level reasoning of what the user wants to do and what the system should do next. User modelling refers to the system's adaptation to the user's personal characteristics, and is often a separate component. However, the knowledge of the user's beliefs, intentions, attitudes, capabilities, and preferences influences the system's decisions on all levels of the interaction management, and can thus be scattered among other components, too.

At the far right, the figure depicts the application interface with some commands needed for managing and manipulating the application-related information. Finally, the architecture also includes various types of static models that encode the system's knowledge of the user, discourse, task, media, etc., and which the system uses in its reasoning processes.

Most current map navigation applications are implemented on distributed architectures (Quickset and OAA, Cohen et al. 1997; SmartKom and Pool, Klüter et al. 2000; HRL and Galaxy, Belvin et al. 200). They allow asynchronous processing of input and thus enable complex information management. Also, modularity supports flexible system development as new components can be integrated or deleted as necessary.

There is also a need for standardising architectures and representations so as to enable seamless technology integration and interaction modelling and also comparison and evaluation among various systems and system components. Recently much effort has been put in standardisations within the W3C consortium which has worked e.g. on the Multimodal Annotation Markup Language EMMA, as well as on standardisation issues concerning adaptation in the system environment. EMMA language is an XML-based markup language for containing and annotating the interpretation of user input. The interpretation of the user's input is expected to be generated by signal interpretation processes, such as speech and ink recognition, semantic interpreters, and other types of processors for use by components that act on the user's inputs such as interaction managers.

9.4.2 Multimodal systems

Research and development on multimodal systems is already more than 25 years old. The first multimodal system is considered to be Bolt's *Put that there* -system (Bolt, 1980), where the users could interact with the world through its projection on the wall and using speech and pointing gestures. The main research goal was to study how actions can disambiguate actions in another modality. Another classic is CUBRICON (Neal and Shapiro, 1988), where the user could use speech, keyboard, and mouse on text, maps, and tables, and the system aim at flexible use of modalities in a highly integrated manner. The QuickSet system (Cohen et al., 1997) is a handheld, multimodal interface for a map-based task, it is been used for extensive investigations concerning pen and speech interface (see e.g. Oviatt, 1997; Oviatt et al., 2000, 2004). Users create entities by speaking their names and distinguishing their properties in the scene, and they can input using speech or pen, or both. Cheyer and Julia (1995) built the TravelMate-system, an agent-based multimodal map application, which has access to WWW-sources, has comparatively rich natural language capabilities, answering to questions such as *Show hotels that are two kilometres from here*. The users can input handwritten, gestural, vocal, and direct manipulation commands, and receive textual, graphical, audio and video data. The navigation system has also been extended to augmented reality.

All these systems investigated possibilities to enhance the system's natural language capabilities with multimodality, and have developed technology for multimodal systems. The German project SmartKom (Wahlster et al., 2001) used a hybrid approach aiming at merging speech, gesture, and graphics technologies into a large system with a general architecture that could be used in developing different applications. The demonstrations were built on three different situations: public information kiosk, home infotainment, and mobile travel companion, and interaction with the system took place via a life-like character Smartakus. On the other hand, the on-going EU project AMI (http://www.amiproject.org/) continues multimodal research in the context of multiparty meeting settings, and aims at developing technology that will augment communications between individuals and groups of people.

Recent mobile systems have effectively tried to capture the technical advancements with mobile devices and location-based services. Robbins et al. (2004) describe map navigation with the help of the ZoneZoom, where zooming lets the user gain an overview and compare information from different parts of the data. The MUMS-system (Hurtig and Jokinen, 2005) allows the user to ask public timetable information using a PDA, and Wasinger et al. (2003) describe a PDA-system that allows users to interact with speech and tactile input, and to get speech and graphical output back. An interesting feature in the system is the fact that the user is allowed to roam the environment, and the system provides extra information when the user asks this as well as spontaneously. The Match system (Johnston et al., 2001), created by AT&T, is a portable multimodal application which allows the users to enter speech and pen gestures to gain access to city help system. Belvin et al. (2001) describe a HRL Route navigation system, while Andrews et al. (2006) and Cheng et al. (2004) discuss generation of route navigation instructions in a multimodal dialogue system.

9.5 Characteristics of multimodal map navigation

Route navigation systems with multimodal interfaces are interactive applications with some particular characteristics due to the specific task dealing with spatial information, location and maps. Moreover, they are technically challenging because of the need to integrate technology, some of which is still under development and not necessarily robust enough for thriving applications. This section focuses on some of these aspects, namely way-finding strategies, cognitive load, and technical aspects. We also discuss a few issues from the mobile navigation point of view.

9.5.1 Wayfinding strategies

It is commonplace that people have different mental maps of their environments and in route guidance situations, it's unlikely that their mental maps coincide. As pointed out by Tversky (2000), people also have erroneous conceptions of spatial relations, especially if they deal with hypothetical rather than experienced environments. People's spatial mental models reflect their conceptions of the spatial world around them, and are constructed in the working memory according to perceptual organizing principles which happen to schematise the environment: for the frequent route navigation tasks this is sensible as there is no reason to remember all the details, but for the infrequent cases, systematic errors occur. In these cases, dialogue capabilities are needed. As discussed in section 9.2, humans are very flexible in providing information that matches the partner's need. The instruction giver's sensitiveness to the particular needs of the information seeker is seen as a sign of cooperation, and in this sense emotional bursts (*it was awful*) and descriptions of apparently unrelated events (*we didn't know which way to continue and nobody knew anything and we had to take the taxi*) provide important information about the underlying task and tacit needs. People also have no trouble in using different viewpoints to describe location information. They can picture the environment from the partner's point of view (*there right when you come out from the metro)*, give abstract route information (*number 79 goes to Malmi but not to the hospital*), navigate an exact route (*so 79 to the station and then 69*), and they can fluently zoom in and out from one place to another (*Hakaniemi, Malmi station*). Verbal grounding of information takes place by references to place names and landmarks (*Hakaniemi, in front of Metallitalo, next to the market place*), by comparison (*close to the end stop*) and familiarity (*change in Hakaniemi if that is more familiar*). Depending on whether the partners are familiar with each other or not their interaction strategies can differ, however. For instance, in the study on the MapTask corpus, Lee (2005) found that conversational participants who were familiar with each other tended to use more complex and a wider range of exchanges than participants who were unfamiliar with each other: the latter tended to use a more restricted range of moves and conformed to direct question-answer type exchanges with more explicit feedback.

In navigation systems, presentation of location information can be approached from different angels and on different abstraction levels, using different perspectives that show the same information but address different user needs and require different output modalities. For instance, Kaasinen et al. (2001) present a list of perspectives commonly used for personal navigation. The list is oriented towards graphical presentations in small screen devices, and it does not take into account the fact that natural language is another coding for the same spatial information, but it shows the variation in styles and possibilities of presentations. First, besides the exact route model, one can also use the schematic model, i.e. an abstraction which only shows the logistic of the route, like a map of underground lines where no distances or specific directions are shown since the information that matters is the final and intermediate points. A graphical representation of the geographical area, usually a map, can be provided by the topological perspective, while a mixture of graphics and language which provides characteristics of the objects and sites on the map as well, is called the topological information view as it allows the user to browse information of the main points of interest.

User experience can be taken into consideration using the egocentric perspectives. The simplest way is to show the location as the users would perceive it if they went to the place themselves; this can be implemented with photographs or 3D-modelling, but also through natural language as exemplified in Dialogue (1). Another perspective, the context-dependent one, also takes the user's interests and attentional state at the current state into account, and aims at answering possible user questions such as "what is close to me?", "what are those?", "what do they want to tell me?" It must obviously match with the user's knowledge and attentional state, and produce natural language descriptions especially tailored to the user's needs.

Although the most natural medium for presenting spatial information is graphics, also natural language in the form of text or speech is used to convey descriptions of the environment: route information can be expressed as a detailed navigation route on the map, but also as a list of route descriptions and instructions. The two perspectives especially using natural language are the guidance and the status perspectives. The former aims at answering questions such as "what needs to be done next? How? When?", and gives the user a set of instructions like "turn at the next crossing" in the form of pictures, text or voice, even in haptic impulses. The latter presents declarative information of the current state of the user, and its output can be pictures, text, or voice, exemplified by "you are here", "there are no changes", "two minutes to your station". The two perspectives correspond to the difference which Habel (2003) has made between navigation commands that deal with actions the user needs to perform and those describing the spatial environment. Imperative commands seem to work better in route planning situations (guidance), while declarative instructions seem more appropriate in real-time navigation (status description).

The use of natural language in route navigation helps the partners to coordinate information and to construct a shared knowledge so as to complete the underlying task. On the other hand, language is also inherently temporal and sequential. When conveying spatial information which is distributed across multiple dimensions and multiple modalities, verbal descriptions are often inaccurate, clumsy, and fragmentary. Tversky (2003) argues that people prefer certain kinds of information to others in specifying spatial location, most notably they prefer landmarks and easy directions, but avoid distance information. People also pay attention to different aspects of the environment: some focus more on visible landmarks, while others orientate themselves according to an abstract representation of the environment. Neurocognitive research shows that there are significant differences in spatial cognition of men and women: men typically outperform women on mental rotation and spatial navigation tasks, while women outperform men on object location and spatial working memory tasks (see summary e.g. in Levin et al., 2005). In the context of route navigation, these findings suggest that the landmark presentation, which requires remembering object locations, seems to work better for female users, while spatial navigation, which is based on the abstraction of the environment and mental representation of space and objects, seems more advantageous to male users.

Although individual performances may differ, the design of a navigation system should take these differences into consideration in its presentation strategies. Research on route descriptions suggests that descriptions that resemble those produced by humans work better with the users than automatically generated list of instructions: the users prefer hierarchical presentation where important parts of the route are emphasised, and also descriptions where salient features of the route are highlighted. This provides a starting point for generating natural route descriptions as in the Coral system (Dale et al., 2005) which produces textual output from raw GIS data using techniques from natural language generation. People also give route descriptions of different granularity: they adapt their instructions with respect to the start and target of the route so that they use more coarse-grained directions in the beginning of the route and change to more detailed descriptions when approaching the target. Using hierarchical representation of the environment, a small set of topological rules and their recursive application, Tomko and Winter (2006) show how granular route directions can be automatically generated, adhering to Gricean maxims of cooperation.

9.5.2 Cognitive load

Humans have various cognitive constraints and motor limits that have impact on the quantity and quality of the information they can process. Especially the capacity of the working memory, seven plus minus two items as originally proposed by Miller (1956) poses limits to our cognitive processes (see also Baddeley (1992) for the multi-component model of the working memory, and Cowan (2001) for an alternative conception of the working memory being part of the long-term memory with the storage capacity of about four chunks). Wickens and Holland (2000) discuss how human perception and cognitive processing works, and provide various examples of how cognitive psychology can be taken into account in designing automated systems. Given the number of possibilities for input/output modalities and the size of screen in hand-held devices, multimodal and mobile interfaces should take the user's cognitive limits into account, and pay attention to the impact of the different modalities on the system's usability as well as on the suitability of each modality for information presentation in terms of cognitive load.

Cognitive load refers to the demands that are placed on a person's memory by the task they are performing and by the distracting aspects of the situation they find themselves in (Berthold and Jameson, 1999). It depends on the type of sensory information, amount of information that needs to be remembered, time and communication limits, language, and other simultaneous thinking processes. For instance, in map navigation, the user is involved in such application-oriented tasks as searching suitable routes, listening navigation instructions, and browsing location information, and in such dialogue-oriented tasks as giving commands and answering questions by the system. Distracting aspects, on the other hand, include external factors like background noise, other people, and events in the user's environment, as well as internal factors like the user being in hurry or under emotional stress.

Cognitive load has an impact on the user's ability to concentrate on the task and thus also on the user's satisfaction with the performance of the system. The effects of overload can be seen e.g. on the features of the speech, weak coordination of the gestures, and in the overall evaluation of the system's usability. Several investigations have been conducted on automatically detecting the symptoms of cognitive load and interpreting them with respect to the user's behaviour (e.g. Berthold and Jameson 1999, Mueller et al. 2001).

Cognitive load is usually addressed by designing the system functionality in a transparent way, by associating communication with clear, simple and unambiguous system responses, and by allowing the user to adjust the design for custom use. It is also important to plan the amount of information given in one go: information should be given in suitable chunks that fit into the user's working memory and current attentional state, and presented in an incremental fashion.

In multimodal interfaces it is also important to consider the use of individual modalities, their relation in communication, and the rendering of information to those modalities that best correspond to the type of information to be delivered. Research concerning multi-media presentations and cognitive load in this respect is large, and interesting observations has been found in relation to human cognitive processing, combination of modalities, and the structure of the working memory. For instance, it is obvious that the cognitive load increases, if the user's attention needs to be split between several information sources, but as shown in Mousavi et al. (1995), visual and auditory modalities also greatly support each other: if information is presented using both visual and auditory modality, the presented items are memorized better than if presented in a single modality. Implications of this line of research can be taken into consideration in multimodal presentations, but their direct application in natural communication or effective integration into design of mobile multimodal systems, such as map navigation, is yet to be resolved. Obviously, this would prove useful. In a recent study, however, Kray et al. (2003) studied different properties of presentation styles in a mobile device, and noticed that cognitive load for textual and spoken presentation is low, but the more complicated visual information is used, or more complicated skills are needed to master the interface, cognitive load rapidly increases and the interface becomes messy and uncomfortable. They also presented guidelines for selecting different presentations depending on the presentation style, cognitive and technical resources and the user's positional information.

It is, however, unclear how to operationalize cognitive load and give guidelines for its measurement in multimodal dialogue systems except by experimenting and trying with different alternatives. The so called "human factors" are taken into account in HCI, but as Sutcliffe (2000) argues, HCI needs a theory, a principled explanation to justify its practise. According to Sutcliffe (2000), the problem is that HCI borrows its theories from cognitive science, but these theories are complex and applied to rather narrow range of phenomena which often fall outside the requirements for practical systems. A solution that would integrate HCI into software engineering would be "design by reuse", i.e. to transfer the knowledge gained in theoretical research into the development of interactive systems, by exploiting patterns that express more generalised specifications and models.

9.5.3 Multimodality and mobility

The notion of mobility brings in the prospects of ambient and ubiquitous computing which provides the users with a new kind of freedom: the usage of a device or a service is not tied to a particular place, but the users can access information anywhere (in principle). Location-based services take the user's location as the starting point and provide services related to this place; route guidance can be seen as one type of location-based service. They are a special case of context-aware systems (Dey, 2001) which provide the user with information or services that are relevant for the task in a given context, and automatically execute the service for the user if necessary. Usually they feature a context-sensitive prompting facility which prompts the user with information or task reminders according to their individual preferences and situational requirements.

Location-based services are a growing service area especially due to mobile and ubiquitous computing scene. Currently there is a wealth of research and development carried out concerning mobile, ubiquitous, and location-aware services. For instance, the EU programme ARTEMIS (*Advanced Research & Technology for Embedded Intelligence and Systems*) focuses on the design, integration and supply of Embedded Computer Systems, i.e. enabling "invisible intelligence" in systems and applications of our everyday life such as cars, home appliances, mobile phones, etc. In the Finnish NAVI-project (Ikonen et al., 2002; Kaasinen et al., 2001), possibilities for building more useful computational services were investigated, and the evaluation of several location-based services brought forward several requirements that the design of such applications should take into account. Most importantly, the design should aim for a seamless solution whereby the user is supported throughout the whole usage situation, and access to information is available right from the point in which the need for that piece of information arises. Integration of route guidance and location-aware services is crucial in this respect: when being guided on the route, the users could receive information about the nearby services and points of interest, and if an interesting place is noted, route guidance to this place is needed.

In mobile situations, the user's full attention is not necessarily directed towards the device, but often divided between the service and the primary task such as moving, meeting people, or the like. The applications thus need to be tailored so that they are easily available when the users want them and when they can use them. This requires awareness of the context which the users are in, so as to support their needs, but intrude them as little as possible. Availability of a service can also be taken into account in the design of the system functionality that should cater relevant usage situations. Concerning route navigation, relevant issues to be taken into consideration in this respect deal with situations when the user is planning a visit (route planning functionality), when they are on the way to a destination (route navigation functionality), and when they are at a given location (way-finding functionality). On the other hand, as noticed by Kaasinen et al. (2001), the situations where mobile location-based systems are used can be demanding. Physical environment causes extra constraints (e.g. background noise and bad illumination can hinder smooth interaction), and in some cases prevent connection altogether (e.g. weather can cause problems with satellite communication).

The requirements for location-aware services in mobile contexts can be grouped as follows (Kaasinen, 2003): comprehensive contents both in breadth (number of services included) and depth (enough information on each individual service), smooth user interaction, personal and user-generated contents, seamless service entities, and privacy issues. We will not go further in these requirements, but note that the three first requirements are directly related to the system's communicative capability and dialogue management. As the computer's access to the context ("awareness") is improved, it can be expected that the type and range of human-computer interaction also changes: it becomes richer, and the need for understanding natural language and use natural dialogues as interface increases. Ideally, the user will be able to conduct rich, adaptive and multimodal communication with the system.

9.5.4 Technical aspects

Fig. 9.2 shows the complexity of multimodal system architecture, and the size of technical integration needed for a working system. Besides the individual components and their functioning, there are also several other aspects that affect the system performance and the user's evaluation of the application. The physical design of the device, the quality of the speech, and the size and precision of the graphical display relate to the system's look and feel, while the overall performance of the system, speed of the data transfer, and the quality and precision of the location contribute to the information processing and the speed of the service.

Location-based services also require that the user can be located. Different positioning systems can be used for this purpose. Satellite positioning enables the user's location to be defined very accurately (2–20 meters), but it does not work indoors and may not work in urban canyons either. The widely used satellite positioning system is GPS (Global Positioning System). European Union has plans to build a corresponding global system called Galileo, which should be in operation in 2008. Using the mobile network, the users can be located via a mobile phone by the network cell in which the phone is located. The cell-based positioning is not very accurate, though: in cities the cell distance is about 50 meters and in rural areas several kilometres, but the advantage of the method is that an ordinary mobile phone is fine and the users need no extra equipment. Large initiatives to reinforce research and development in mobile and wireless communications, mobile services and applications also exist: e.g. the European technology platform eMobility (www.emobility.eu.org) aims at establishing relations with research programmes, paving the way for successful global standards.

9.6 An example: the MUMS-system

MUMS (MUltiModal route navigation System) has been developed within the technology project PUMS, cooperation among Finnish universities and supported by the Finnish Technology Agency TEKES and several IT-companies. It is based on the Interact-system (Jokinen et al. 2002) which aimed at studying methods and techniques for rich dialogue modelling and natural language interaction in situations where the interaction had not been functional or robust enough. The technical development in MUMS has been directed towards the integration of a PDA-based graphical and tactile interface with speech input and output. Also possibilities of exploiting GPS information of the user's current location in an interactive navigation system have been explored, but currently the MUMS-system and the GPS-based location system work independently. The main goal of the project is to build a robust practical application that provides real-time travel information and mobile navigation for visitors to Helsinki. The system is planned to be demostrated in the international UITP Mobility and City Transport conference to be held in Helsinki in 2007.

9.6.1 Example interaction

An interface to a location-based service must be able to provide different information and several functionalities to the user. Functional requirements for a multimodal route navigation system can be summarised as follows. The system should provide the user with

- off-line route planning (comparison, evaluation of alternatives)
- on-line route guidance
- ex-tempore exploration of location
- access to location-based data (bus routes, points of interest, restaurants, etc.)
- helpful information about the use of the system

The MUMS system can perform the two basic tasks: provide timetable information for public transportation and provide navigation instructions for the user to get from a departure place to a destination. As the database only contains bus routes, exploration of the location or search for various points of interest is not supported in the first system version. The user help is currently implemented via a text-based menu.

As shown in the example dialogue (4), the users can supply information either by voice or a map gesture (i.e. pointing the map with a pen); they can correct or change the parameter values, and also iterate departure times until a suitable route is found. The system fetches the route details from the route database, but it can also take the initiative and prompt the user for missing information, if either the departure or arrival location is unknown. The system can also provide information on the different legs of the route, so that the users can get situation-tailored navigation online. The spoken summary contains the time of departure, the form of transportation, the line number (where applicable), the number of changes from a vehicle to another, the name or address of the final destination, and the time of arrival.

Dialogue (4)

U: Uh, how do I get from the Railway station ... uh…
S: *Where was it you wanted to go?*
U: Well, there! + <map gesture>
S: *Tram 3B leaves Railway Station at 14:40, there is one change. Arrival time at Brahe Street 7 is 14:57.*
U: When does the next one go?
S: *Bus 23 leaves Railway Station at 14:43, there are no changes. Arrival time at Brahe Street 7 is 15:02.*
U: Ok. Navigate.
S: *Take bus 23 at the Railway Station at 14:43.*
U: Navigate more.
S: *Get off the bus at 14:49 at the Brahe Street stop.*
U: Navigate more.
S: *Walk 200 meters in the direction of the bus route. You are at Brahe Street 7.*
U: Okay.

Fig. 9.3 shows the map of Helsinki city centre and the location of the user’s map gesture (a circle). When the route information has been fetched from the database, the route is presented on the screen (Fig. 9.3, at right), and details as shown in the dialogue above, are also supplied by speech. During navigation the route and the spoken details are available, and also a text-based instruction list is available via a special menu button, so as to allow the user to check the route quickly and discreetly.

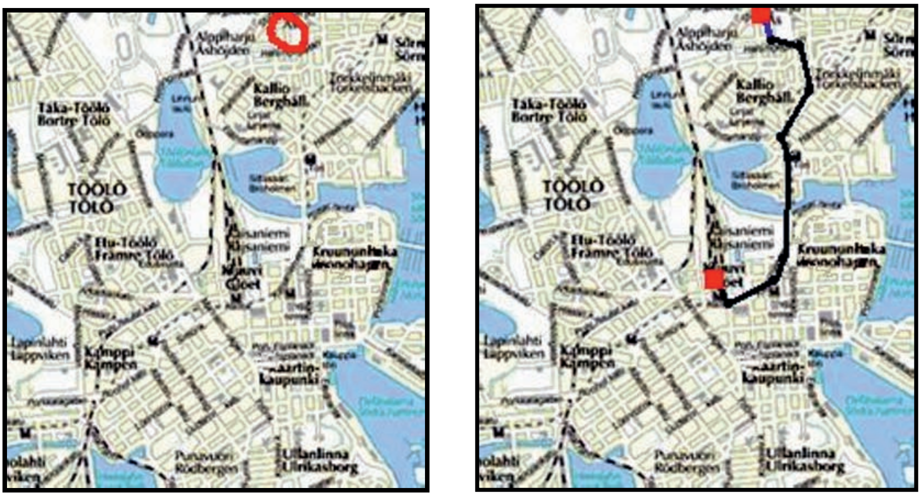


Fig. 9.3. Tactile user input (at left) and a graphical representation of a route (at right).

The system design aims at natural dialogue capabilities that would exhibit adaptation with respect to various users, and e.g. navigation can be started either in a concise or detailed mode, depending on the user’s needs. The system’s presentation strategies take into account chunk-wise presentation (Cheng et al., 2004): given that the system has a long list to present to the user, the items are presented in chunks that are of reasonable size. If the list contains items which may be unknown to the user or which may be of great importance, the size of the chunks is adapted accordingly.

The system’s dialogue strategies concern confirmation and initiative strategies. The former deal with the system leaving uncertainty of the user input to be resolved later in the dialogue. The supporting nature of multimodality comes to play a role here, since the uncertainty can be resolved with the help of pen-gesture, and thus the interaction becomes more natural and less annoying for the user. The initiative strategy is used in the navigation task where the user is guided through the route. It is envisaged that in these situations, the users may be allowed to interrupt and enter into “explorative” mode, if they want to explore the environment more (Wasinger et al., 2003).

9.6.2 System architecture

A general outline of the system architecture is given in Fig. 9.4. The system consists of a PDA client device and a remote system server. The server handles all processing of the user-provided information, and the PDA only has a light-weight speech synthesizer. The PDA can be considered a user interface: it accepts speech and tactile input and presents information via speech and graphical map data. The touch-screen map interprets all tactile input as locations: a tap on the screen denotes a pinpoint coordinate location, whereas a circled area will be interpreted as a number of possible locations. The map can be freely scrolled and zoomed in real time, and the inputs are recorded simultaneously and time stamped for later modality fusion phase processing.

We use the Jaspis architecture (Turunen and Hakulinen, 2003), which is a distributed and modular platform designed originally for speech systems. The system consists of different modules which take care of the different phases of processing (input analysis, dialogue management, and presentation), and via the Task Handler module, it is connected to an external route finder system and database, which returns, for each complete query, a detailed set of route information in XML format. The route information is stored in a local database so it is easily available for creating summaries and providing the user with detailed route information.

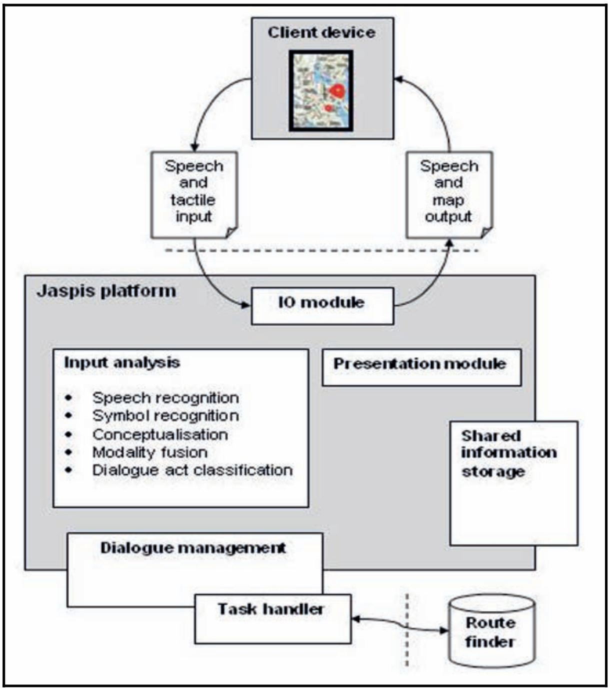


Fig. 9.4. General architecture of the MUMS-system (Hurtig and Jokinen, 2006).

The processing of the user input proceeds in three steps. It begins with the recognition of each modality and the attaching of high-level task-relevant concepts, e.g. "explicit_speech_location" to input units. Then the fusion of modalities takes place and results in an N-best list of user input candidates. The final phase attaches a dialogue act type to each of the fused inputs. The process then advances to the dialogue management module which has access the dialogue history. The dialogue manager determines user intentions and chooses the input candidate that best fits the situation and task at hand, and then carries out the corresponding task. Depending on the content of the user input and the state of the dialogue, the dialogue management module forms a generic response which is then further specified by the presentation module. The presentation module formats the response according to the user preferences and the client hardware in use, and sends the information to be presented in the client device.

9.6.3 Multimodal fusion

Technical setup for the coordination and combination of modalities is complex. Multiple modalities can be combined in different ways to convey information. Martin (1997) identifies the following relations between modalities: equivalence (modalities can convey the same information), specialization (a modality is used for a specific subset of the information), redundancy (information conveyed in modalities overlaps), complementarity (information from different modalities must be integrated in order to reach coherent information), transfer (information from one modality is transferred to another), concurrency (information from different modalities is not related, but merely speeds up interaction). The consequences of the different alternatives on the interpretation and planning of information exchanges, and on the processing architecture of the system needs to be investigated more.

The process of combining inputs from different modalities to create an interpretation of composite input is called as multimodal fusion. Nigay and Coutaz (1993, 1995), cf. also Gibbon et al. (2000), have identified three levels of fusion: lexical, syntactic, and semantic. The lexical fusion happens when hardware primitives are bound to software events (e.g. selecting objects when the shift-key is down), the syntactic fusion involves combining data to form a complete command, and semantic fusion specifies the detailed functionality of the interface and defines the meanings of the actions. They also present a fusion architecture called PAC Amodeus which supports concurrent signal processing and use a melting pot -metaphor for syntactic and semantic fusion: information from the lower level is first combined into higher abstractions, from which the second level fusion obtains a complete command and sends it to the functional components. Other architectures, like OAA, have a special multimodality coordination agent (MC agent) which produces a single meaning from multiple input modalities aiming at matching this with the user's intentions. It searches the conversation context for an appropriate referent, and if this cannot be found, a question to the user is formulated. Before asking, the agent waits for a short time so as to allow synchronization of the different input modalities to take place.

In MUMS, the fusion of the inputs takes place in three phases, see Fig. 9.5. They correspond to following operations:

1. production of legal combinations,
2. weighting of possible combinations,
3. selection of the best candidate.

In the first phase, speech and tactile data are paired so that their order of occurrence is preserved. All possible input combinations for observed graphical symbols and language concepts are thus collected. In the example above, there are three possible combinations which maintain the order of input: {pointing_1, *Railway station*}, {pointing_2, *Railway station*}, {pointing_2, *from the Operahouse*}. In the second phase, the weight of each concept-symbol pair is calculated. The weighting is based on parameters that deal with the overlap, proximity, as well as with the quality and type of the fused constituents. Overlap and proximity concern temporal qualities of the concept and the symbol: e.g. Oviatt et al (2004) regard temporal proximity as the single most important factor in combining constituents in speech and tactile data. The concept type refers to the linguistic properties of the concept: it can be a pronoun (*here*), a noun phrase indicating location (*Railway Station*), or an implicitly named location, such as an address (*Brahe Street 7*), and a location gesture. The weighting is carried out for each fused pair in each input candidate, based on which the candidate is then assigned a final score. An N-best list of these candidates is then passed on to the third phase.

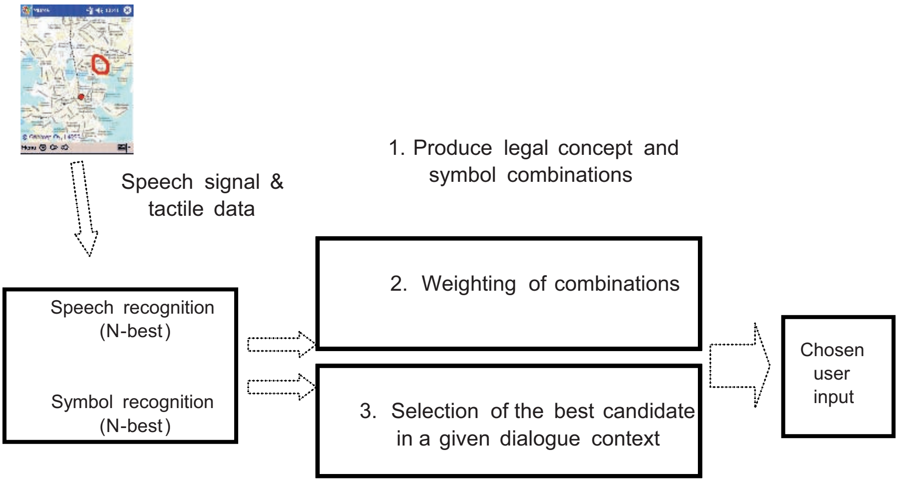


Fig. 9.5. Graphical presentation of the MUMS multimodal fusion (from Hurtig and Jokinen 2006).

On the third and final phase, dialogue management attempts to fit the best ranked candidates to the current state of dialogue. If a candidate makes a good fit, it is chosen and the system will proceed on formulating a response. If not, the next candidate in the list will be evaluated. Only when none of the candidates can be used, the user will be asked to rephrase or repeat his/her question. A more detailed description of the fusion component can be found in Hurtig (2005).

9.6.4 Evaluation

Evaluation of multimodal systems is problematic not only because of the many components but also because of the lack of methodology. Evaluation frameworks for speech-based services like that presented by Möller (2001) can be extended to multimodal systems too, but in general, it is difficult to assess system performance and usability in objective terms, or provide clear metrics for system comparison.

The target group for the MUMS-system is mobile users who quickly want to find their way around. The first evaluation of system was conducted among 20 users who participated in four scenario-based route navigation tasks. The scenarios dealt with different situations where the users are supposed to find the best way to meetings, theatre, and restaurants. The scenarios were designed so that they would most likely support speech interaction (e.g. location not on the map shown on the PDA), tactile interaction (e.g. exact place), or be neutral between the interaction modality. The participants were recruited through calls in the internal mailing lists and consisted of students, senior researchers, and employees from the Helsinki City Transportation Agency, aged between 20 and 60. Most of them had some familiarity with the interactive systems, but no experience with the PDA, or the MUMS system.

Evaluation focused especially on the user experience, and comparison of the users' expectations and real experience concerning the system properties and functionality. For this purpose the participants were asked to rate their expectations about the system before the task commenced, and also give their opinions of the system performance after the tasks. The influence of the user's preconceptions of the system was tested by dividing participants into two groups, so that the other half of the participants were told they were testing a speech-based interface which is extended with a multimodal map, and the other half was told they were testing a tactile interface which has also speech facility.

Results are reported in (Jokinen and Hurtig, 2006) and here we only give a brief outline. In general, the users gave positive comments about the system which was regarded as new and fascinating. As expected, speech recognition caused problems such as unrecognized commands and repetition of questions, which were considered a nuisance. However, all test users were unanimous that the system with both speech and tactile possibilities is preferable to a system with unimodal input and output. This agrees with Oviatt et al. (1997) who found that multimodal input was preferable to unimodal commands, although there were differences in cases where the users preferred multimodal (speech+pen) to speech input. Experimental evaluation of the Match-system (Johnston et al., 2001) also shows similar tendency: although unimodal pen commands were recognized more successfully than spoken commands, only 19% of the interactions were pen-only; more than half of the exchanges were conducted by speech, and about 28% were multimodal. On the other hand, it's not clear if the statistics is due to the users preferring speech to tactile input, or to the particular commands needed in the tasks favouring speech rather than tactile modality.

It was interesting that the tactile group evaluated the system's speech-based interaction much higher than the speech group. They also considered the system faster, although the system performance was the same. These differences are obviously due to the expectations that the users had about the system. The tactile group regarded tactile input as the primary mode of interaction and the fact that they could also talk to the system was apparently an interesting additional feature which should not be assessed too harshly. In other words, speech provided some extra value for the tactile group. The speech group, however, expected the task to be completed via speech and had high demands for this primary mode of their interaction. Although tactile input was considered new interesting technology, its use might have been regarded as a fallback in cases where the primary mode of communication failed.

Significant differences were also found concerning the users' age. Younger users found the system of higher quality and better value in general than the older users, but interestingly, the age group between years 33 and 48 had very low pre-test expectations about the system's performance and usability, although their opinions were brought up to the same level as the younger and older users in the post-task evaluation. As for gender differences, differences were found only in that the female users assessed the system's "softer" characteristics more positively than the male users.

The results show that individual users perceive and value multimodal dialogue systems differently. Although the differences in evaluation may not always be pinpointed down to a single aspect such as prior knowledge, predisposition, age, or gender differences, it is important to notice that the results seem to support adaptive and flexible system design, where various users and their preferences should be taken into account and multimodal options would give the users freedom to choose between different interaction styles.

9.7 Discussion and future research

In this chapter we have discussed various aspects concerning multimodality, multimodal mobile systems, route navigation, and location-based services. We have also given a short outline of the MUMS-system, a multimodal route navigation system that combines both speech and tactile IO-capabilities, and discussed results of the user evaluation. The results show that the users' predisposition towards the system has influence on their evaluation of the system: prior expectations about the system's capabilities, especially concerning different modalities, change the users' perception of the system and its usability. Also the newness factor plays a part, since novice users tend to be fascinated by the novel aspects of the system, whereas more experienced users tend to emphasize the system's usability. Speech also puts high demand on the system's verbal communication fluency, and adds an extra difficult aspect to system evaluation: the users easily expect the system to possess more fluent spoken language capabilities than what is technologically possible.

The system will be extended to handle more complex pen gestures, such as areas, lines and arrows. As the complexity of input increases, so does the task of disambiguation of gestures with speech. Temporal disambiguation has also been shown to be problematic; even though most of the time speech precedes the related gesture, sometimes this is not the case. The integration and synchronisation of information in multimodal dialogue systems is thus a further research topic. It is also important to study appropriate modality types for the different tasks. Although multimodality clearly improves task completion, the enhancement seems to apply only on spatial domains, and it remains to be seen what kind of multimodal systems would assist in other, more information-based domains. This is related to the use of different perspectives and selection of presentations that would best fit to the task and the individual user's preferences.

The PUMS-project also aims at addressing Design for all –principles, i.e. to provide users with special needs with a better access to digital information and assistance in their everyday tasks. The Association for Visually Impaired People is a project partner, and we expect various navigation solutions to be further improved through the feedback from the visually impaired users. Enhanced natural language communication capabilities are also called for in real time navigation. However, an interesting concern was raised in the discussions: apparently the quality of speech synthesizer should not be too human like since the partner should be identifiable as a machine or a human. Real-time positioning of the user by GPS is also a challenge and the project continues in this direction too. It is likely that the integration of positioning and map interaction will take place later in the project when both components have been consolidated enough.

Maybury and Wahlster (1998) list several advantages for multimodal interaction. The main benefit seems to be the increased naturalness of communication compared with single-modality systems: the users can exploit the same interaction styles and strategies that they use in human-human communication, and thus interaction becomes more intuitive and natural, and presumably also easier. Gibbon et al. (2000) point out that the users may also have different modality preferences, and multimodal systems offer freedom of choice, thus contributing to more flexible and enjoyable interaction. Multimodal systems also enable more efficient interaction. For instance, Oviatt et al. (1997) observed that the task was completed more rapidly and efficiently when using complementary pen and speech modalities than a single modality. Also interpretation accuracy can be boosted with the help of redundant and/or complementary modalities; e.g. in noisy environments, it is often beneficial to combine speech recognition and lip-reading (Yang et al. 1998). Maybury and Wahlster talk about cross-modal synergy: one communication channel can help to refine inaccuracies and resolve ambiguities in another channel. From the usability point of view, different modalities have different benefits which contribute to the ease of use and usefulness of system: e.g. it is easier to point to an object than refer to it by a verbal description, and to maintain privacy, graphical interface is preferred to a speech-based one.

There are also several challenges in developing interactive systems that could offer natural and rich multimodal interaction. Multimodal interfaces require multidisciplinary expertise and engineering efforts: better understanding of natural communication and choice of modalities for efficient presentation. For mobile applications, adaptation needs to be addressed: the system will be used in different environments, by various users with different abilities and requirements.

We must also remember that the users want solutions which would help them to make their life easier. The goal of "doing the right thing at the right time" is usually regarded as the core of the effective interaction, but the ultimate requirement of effectiveness can also be seen as bad: in the user studies conducted by Kaasinen (2003) on location-aware systems, the participants pointed out that some usage scenarios created a feeling of haste, and that there is a danger that the servant becomes a master that starts to give commands to the user.

In the design and development of location-based services such as multimodal route navigation, care must be taken to avoid narrow use of new technology and to allow the users to interact with the system in a way that is natural to them. This chapter has argued that this kind of naturalness cannot be reached by studying efficient task completion only; rather, the system design should pay attention to cooperative interaction and the system be equipped with appropriate natural language communication capabilities. As the users' perception of interactive systems and their assessment of the usability of the applications seem to depend on the systems' communicative fluency, the combination of multimodal dialogue strategies with comprehensive natural language content might thus be a good starting point for building better location-based services.

Acknowledgements

This work has been conducted in the Finnish National Technology Agency TEKES supported framework Fenix, as part of the collaboration project PUMS (New Methods and Techniques of Speech Technology). I would like to thank the project partners for collaboration in the project, and especially Topi Hurtig for discussions on multimodal maps and route navigation as well as for the realization of the MUMS-system.

References

Allwood, J. (1976): *Linguistic Communication as Action and Cooperation*. Department of Linguistics. University of Gothenburg. Gothenburg Monographs in Linguistics 2.

Andrews, T., Broughton, M., Estival, D. (2006): Implementing an Intelligent Multimedia Presentation Planner using an Agent Based Architecture. In *Procs. of the Workshop on Multimodal Dialogues. Int Conf on Intelligent User Interfaces*, Sydney, Australia.

Baddeley, A.D. (1992): Working Memory. *Science,* 255:556—559.

Belvin, R., Burns, R., Hein, C. (2001): Development of the HRL Route Navigation Dialogue System. In *Procs of the 1st Int Conf on Human Language Tech Research*, paper H01-1016.

Berthold, A., Jameson, A. (1999): Interpreting symptoms of cognitive load in speech input. In Kay, J. ed. *Procs of the 7th Int Conf on User Modeling (UM99),* Springer, Wien, 235-244.

Bolt, R.A. (1980): Put-that-there: Voice and gesture at the graphic interface. *Computer Graphics*, 14(3): 262-270.

Cheng, H., Cavedon, L., Dale, R. (2004): Generating Navigation Information Based on the Driver's Route Knowledge. In Gambäck, B. and Jokinen, K. eds. *Procs of the DUMAS Final Workshop Robust and Adaptive Information Processing for Mobile Speech Interfaces*, COLING Satellite Workshop, Geneva, Switzerland, 31–38.

Cheyer, A., Julia, L. (1995): Multimodal Maps: An Agent-based Approach. In *Procs of Int Conf on Cooperative Multimodal Communication (CMC/95),* Eindhoven, The Netherlands.

Clark, H., Schaefer, E.F. (1989): Contributing to Discourse. *Cognitive Science,* 13:259-294.

Clark, H., Wilkes-Gibbs, D. (1986): Referring as a collaborative process. *Cognition,* 22:1--39.

Cohen, P.R., Johnston, M., McGee, D.R., Oviatt, S.L., Pittman, J., Smith, I., Chen, L., Clow, J. (1997): QuickSet: Multimodal interaction for distributed applications. *Procs of the 5th Int Multimedia Conf (Multimedia '97)*, ACM Press, Seattle, WA, 31—40.

Cohen, P.R., Levesque, H.J. (1991): Teamwork. *Nous*, 25(4):487-512.

Cowan, N. (2001): The Magical Number of 4 in Short-Term Memory: A Reconsideration of Mental Storage Capacity. *Behavioural and Brain Sciences,* 24(1): 87-185.

Dale, R., Geldof, S., Prost, J-P. (2005): Using Natural Language Generation in Automatic Route Description. *Journal of Research and Practice in Information Technology*, 37(1): 89-105.

Danieli M., Gerbino E. (1995): Metrics for Evaluating Dialogue Strategies in a Spoken Language System. *Working Notes, AAAI Spring Symposium Series*, Stanford University.

Dey, AK (2001): Understanding and using context. *Personal and Ubiquitous Computing* 5:20–24

Gibbon, D., Mertins I.., Moore R., (Eds.) (2000): *Handbook of Multimodal and Spoken Dialogue Systems; Resources, Terminology, and Product Evaluation*. Kluwer Academic Publishers.

Granström, B., (Ed). (2002): *Multimodality in Language and Speech Systems*. Dordrecht: Kluwer.

Grice, H.P. (1975): Logic and Conversation. In Cole, P. and Morgan, J.L. eds. *Syntax and Semantics. Vol 3: Speech Acts*. Academic Press.

Grosz, B., Sidner, C.L. (1990): Plans for discourse. In Cohen, P.R, Morgan, J. and Pollack M.E. eds. *Intentions in Communication*. MIT Press.

Habel, C. (2003): Incremental Generation of Multimodal Route Instructions. In *Natural Language Generation in Spoken and Written Language, AAAI Spring Symposium*. Palo Alto.

Harnard, S (1990): The Symbol Grounding Problem. *Physical*, D 42: 335-346.

Heeman, P. A., Hirst, G. (1995): Collaborating on referring expressions. *Computational. Linguistics.* 21(3): 351-382.

Hurtig, T. (2005): Multimodaalisen informaation hyödyntäminen reitinopastusdialogeissa. (”Utilizing Multimodal Information in Route Guidance Dialogues”). Helsinki University of Technology. Department of Electrical and Communication Engineering.

Hurtig, T., Jokinen, K. (2005): On Multimodal Route Navigation in PDAs. In *Procs of the 2nd Baltic Conf on Human Language Technologies*, Tallinn, Estonia, 261-266.

Hurtig, T., Jokinen, K. (2006): Modality fusion in a route navigation system. In *Procs of the IUI 2006 Workshop on Effective Multimoda Dialoguel Interfaces*, 19-24

Ikonen, V., Anttila, V., Petäkoski-Hult, T., Sotamaa, O., Ahonen, A., Schirokoff, A., Kaasinen, E. (2002): Key Usability and Ethical Issues in the NAVI programme (KEN). Deliverable 5. Adaptation of Technology and Usage Cultures. Part 1. Version 1.2. Available at: http://www.naviverkosto.org/cd/ken_d5_i_12.pdf

Johnston, M. (1998): Unification-based multimodal parsing. *Procs of the 36th Annual Meeting on Association for Computational Linguistics*, 624-630, Montreal, Canada.

Johnston, M., Cohen, P.R., McGee, D., Oviatt, S., Pittman, J. and Smith, I. (1997): Unification-based multimodal integration. In *Procs of the 8th Conf on European Chapter of the Association for Computational Linguistics*, 281-288, Madrid, Spain.

Johnston, M., Bangalore, S., Vasireddy, G., Stent, A., Ehlen, P., Walker, M., Whittaker, S., Maloor, P. (2002): MATCH: an architecture for multimodal dialogue systems. *Procs of the 40th Annual Meeting on Association for Computational Linguistics,* Philadelphia, 376-383.

Jokinen, K. (2004): Communicative Competence and Adaptation in a Spoken Dialogue System. *Procs of the 8th Int Conf on Spoken Language Processing (ICSLP)* CD-Rom Jeju, Korea.

Jokinen, K., Hurtig, T. (2006): User Expectations and Real Experience on a Multimodal Interactive System. *Procs of Interspeech-2006*, Pittsburgh, US.

Jokinen, K., A. Kerminen, M. Kaipainen, T. Jauhiainen, G. Wilcock, M. Turunen, J. Hakulinen, J. Kuusisto, K. Lagus (2002): Adaptive Dialogue Systems: Interaction with Interact. In

Jokinen, K., and McRoy, S. eds. *Procs of the 3rd SIGDial Workshop on Discourse and Dialogue*, Philadelphia, USA, 64 – 73.

Jokinen, K., Raike, A. (2003): Multimodality – technology, visions and demands for the future. *Procs of the 1st Nordic Symposium on Multimodal Interfaces*, Copenhagen, Denmark.

Kaasinen, E. (2003): User needs for location-aware mobile services. *Personal and Ubiquitous Computing*, 7:70–79.

Kaasinen, E., Luoma, J., Penttinen M., Petäkoski-Hult, T., Södergård, R. (2001): Basics of Human-Centered Design in Personal Navigation, NAVI programme report. Available at: http://www.naviverkosto.org/cd/ken_d1_10.pdf

Klüter, A., Ndiaye, A., Kirchmann, H. (2000): VerbMobil from a Software Engineering Point of View: System Design and Software Integration. In Wahlster, W. ed. *VerbMobil: Foundations of Speech-to-Speech Translation*, Springer, Berlin.

Kray, C., Laakso, K., Elting, C., Coors, V. (2003): Presenting route instructions on mobile devices. In *Procs of IUI 03*, 117—124. Miami Beach, FL, ACM Press.

Lee, A. (2005): The Effect of Familiarity on Knowledge Synchronisation. In Heylen, D. and Marsella, S. eds. Mind Minding Agents. AISB Symposium on Mind-Minding Agents (track of Virtual Social Agents), University of Hertfordshire, Hatfield, England.

Levin, S. L., Mohamed, F. B. and Platek, S. M. (2005): Common ground for spatial cognition? A behavioral and fMRI study of sex differences in mental rotation and spatial working memory. *Evolutionary Psychology, 3*: 227-254.

Maaß, W. (1995): From Visual Perception to Multimodal Communication: Incremental Route Descriptions. In Mc Kevitt, P. ed. *Integration of Natural Language and Vision Processing: Computational Models and Systems,* Vol. 1, 68-82. Kluwer, Dordrecht.

Maaß, W., Wazinski, P., Herzog, G. (1993): VITRA GUIDE: Multimodal Route Descriptions for Computer Assisted Vehicle Navigation. In *Procs of the 6th Int Conf on Industrial and Engineering Applications of Artificial Intelligence and Expert Systems IEA/AIE-93*, 144—147. Edinburgh, Scotland.

Martin, J.-C. (1997): Towards `intelligent' cooperation between modalities: the example of multimodal interaction with a map. In *Procs of the IJCAI'97 Workshop on Intelligent Multimodal Systems*.

Maybury, M.T., (Ed.) (1993): *Intelligent Multimedia Interfaces*. AAAI Press, Menlo Park, CA.

Maybury, M. T., Wahlster, W. 1998, eds. *Readings in Intelligent User Interfaces*. Morgan Kaufmann, San Francisco, CA.

Miller, G. A. (1956): The magical number seven, plus or minus two: Some limits on our capacity for processing information. *Psychological Review*, 63: 81-97.

Mousavi, S.Y., Low, R., Sweller, J. (1995): Reducing Cognitive Load by Mixing Auditory and Visual Presentation Modes. *Journal of Educational Psychology,* 87:2, 319-334.

Mueller, C. Großmann-Hutter, B., Jameson, A., Rummer, R., Wittig, F. (2001):Recognizing time pressure and cognitive load on the basis of speech: An experimental study. In Bauer, M., Vassileva, J., Gmytrasiewicz, P. eds. *Procs of the 8th Int Conf on User Modeling (UM2000)*, 24–33. Springer, Berlin.

Möller, S. (2002): A New Taxonomy for the Quality of Telephone Services Based on Spoken Dialogue Systems. In Jokinen, K and McRoy, S. eds. *Procs of the 3rd SIGDial Workshop on Discourse and Dialogue*, Philadelphia, USA.

Neal, J.G., Shapiro, S.C. (1991): Intelligent Multi-media Interface Technology. In Sullivan, J.W., Tyler, S.W. eds. *Intelligent User Interfaces*, Frontier Series, ACM Press, New York 11-43.

Nigay, L., Coutaz, J. (1995): A generic platform for addressing the multimodal challenge. *Procs of ACM-CHI'95 Conf on Human Factors in Computing Systems*, ACM Press, 98-105.

Oviatt, S. L. (1997): Multimodal interactive maps: Designing for human performance. *Human-Computer Interaction Special issue on Multimodal Interfaces*, 12: 93-129

Oviatt, S., De Angeli, A., Kuhn, K. (1997): Integration and synchronization of input modes during multimodal human-computer interaction. *Human Factors in Computing Systems (CHI'97)*, 451-422, ACM Press, New York.

Oviatt, S., Cohen, P.R., Wu, L., Vergo, J., Duncan, L., Suhm, B., Bers, J., Holzman, T., Winograd, T., Landay, J., Larson, J., Ferro, D. (2000): Designing the User Interface for Multimodal Speech and Pen-based Gesture Applications: State-of-the-Art Systems and Future Research Directions. *Human Computer Interaction*, 15(4): 263-322.

Oviatt, S., Coulston, R., Lunsford, R. (2004): When Do We Interact Multimodally? Cognitive Load and Multimodal Communication Patterns. In *Procs of the 6th Int Conf on Multimodal Interfaces (ICMI 2004)*, Pennsylvania, USA.

Robbins, D., Cutrell, E., Sarin, R., Horvitz, E. (2004): ZoneZoom: Map Navigation for Smartphones with Recursive View Segmentation. In *Procs of AVI*.

Seneff, S., Hurley, E., Lau, R., Pao, C., Schmid, P., V. Zue (1998): GALAXY-II: A reference architecture for conversational system development. *Procs of ICSLP 98*. Sydney, Australia.

Shepherd, G.M. (1988): *Neurobiology*. Oxford University Press, Oxford.

Stenning, K., Oberlander, J. (1995): A cognitive theory of graphical and linguistic reasoning: logic and implementation, *Cognitive Science*, 19: 97-140.

Sutcliffe, A. (2000): On the effective use and reuse of HCI knowledge. *ACM Transactions on Computer-Human Interaction*, Special Issue on Human-Computer Interaction in the New Millennium, Part 2, 7(2):197 – 221.

Tomko, M., Winter, S. (2006): Recursive Construction of Granular Route Directions. *Journal of Spatial Science*, 51(1): 101-115.

Turunen, M., Hakulinen, J. (2003): Jaspis 2 - An Architecture For Supporting Distributed Spoken Dialogues. In *Procs of the Eurospeech 2003*, 1913—1916.

Tversky, B. (2000): Multiple Mental Spaces. Plenary talk at the Int Conf on Rationality and Irrationality, Austria. Available at: www-psych.stanford.edu/~bt/space/papers/rationality.pdf

Tversky, B. (2003): Places: Points, Planes, Paths and Portions. In van der Zee, E. and Slack, J. eds. *Representing Direction in Language and Space*. Oxford University Press. 132-143.

Vernier, F., Nigay, L. (2000): A Framework for the Combination and Characterization of Output Modalities. In *Procs of 7th Int Workshop on Design, Specification and Verification of Interactive Systems (DSV-IS'2000),* Limerick, Ireland, 32-48.

Wahlster, W., Reithinger, N., Blocher, A. (2001): SmartKom: Multimodal Communication with a Life-I. Character. In *Procs of Eurospeech2001*, Aalborg, Denmark.

Walker, M. A., Langkilde, I., Wright, J., Gorin, A., Litman, D.J. (2000): Learning to Predict Problematic Situations in a Spoken Dialogue System: Experiments with How May I Help You? In *Procs of NAACL'00*, Seattle, US, 210-217.

Wasinger, R., Oliver, D., Heckmann, D., Braun, B., Brandlarm, B., Stahl, C. (2003): Adapting Spoken and Visual Output for a Pedestrian Navigation System, based on given Situational Statements. *Workshop on Adaptivity and User Modelling in Interactive Software Systems (ABIS)*, 343-346.

Wickens, C.D., Holland (2000): *Engineering Psychology and Human Performance*. Prentice-Hall, Upper Saddle River, NJ.

Yang, J., Stiefelhagen, R., Meier, U., Waibel, A. (1998): Visual tracking for multimodal human computer interaction. Human Factors in Computing Systems, In *Procs of CHI'98*, ACM Press, 140-147.

Yankelovich, N. (1996): How do users know what to say? *Interactions,* 3(6): 32-43.

Websites:

Dagstuhl Seminar on Multimodality	www.dfki.de/~wahlster/Dagstuhl_Multi_Modality/
EMMA Extensible MultiModal Annotation Markup Language	www.w3.org/TR/emma/
Interact-project	www.mlab.uiah.fi/interact/
ISLE/NIMM reports	isle.nis.sdu.dk/reports/wp11/
MIAMI Multimodal definitions	www.ai.rug.nl/~lambert/projects/miami/taxonomy/taxonomy.html
MUMIN network	www.cst.dk/mumin
MUMS	www.helsinki.fi/~thurtig/MUMS/
W3C Recommendation	www.w3.org

10 Designing Interactions for Navigation in 3D Mobile Maps

Antti NURMINEN[1], Antti OULASVIRTA[2]

[1]Helsinki University of Technology, Finland
[2]Helsinki Institute for Information Technology (HIIT), Finland

Abstract. Due to their intuitiveness, 3D mobile maps have recently emerged as an alternative to 2D mobile maps. However, designing interactions for navigation in a 3D environment using a mobile device is non-trivial. Challenges are posed by the severe limitations of the mobile user interface and of the capacities of the mobile user. This chapter analyses *the problem of degrees of freedom*: how to make navigation quicker and more intuitive by the means of restricting and guiding movement, yet enabling unrestricted access to all reasonable points-of-interests. Insights from empirical studies of mobile map interaction are presented, in the form of a *model* of interactive search, to draw requirements for interaction design. Then, the design of controls, landmarks, cameras, interest fields, routes, paths etc. are analysed and several higher-level navigation metaphors are discussed. We propose ways to support spatial updating, rapid alignment of physical and virtual spaces, and overcoming the keyhole problem. A working prototype system is used to illustrate different solutions alongside with alternative designs, weighing their pros and cons.

10.1 Introduction

Recent advances in the processing capabilities and interface technologies of mobile devices have brought about a situation where 3D mobile maps are increasingly realistic. During the past five or so years, several developers have presented working prototypes of 3D mobile maps and empirically compared them to 2D maps. Various user benefits have been identified, such as fun, recognisability, efficiency, intuitiveness, and decreased memory load (e.g., Burigat and Chittaro, 2005; Laakso 2002, Oulasvirta, Nurminen, and Nivala, submitted; Rakkolainen and Vainio, 2001; Vainio and Kotala, 2002).

Fig. 10.1 presents 2D and 3D maps of the same area. There are important differences between 2D and 3D mobile maps in the way each supports orientation and navigation. Particularly, orientation with 2D maps (electronic or paper) requires identifying possibly abstract cues, symbols, and shapes of a map as well as real world objects, and performing a mental transformation between them. A 3D map can provide a more directly recognisable visualisation of the environment. With a first person view, even the mental transformation would become unnecessary. While a 3D representation seems to provide a very intuitive static view of the environment, (interactive) navigation in such a virtual environment is more complex. Consequently, while 3D visualisations appear as an appealing option for designers, a major problem lies within the meaningful and efficient control over such visualisations.

Furthermore, there are non-trivial design problems arising from the characteristics of mobile human-computer interaction (HCI). The key challenge in design is the problem of *degrees of freedom* (DOFs): how to balance freedom of movement with efficiency of navigation. The number of DOFs needed to completely control a 3D view exceeds the amount of input controls in mobile devices, and the problem is accentuated by the fact that mobile devices have small displays, which implies that more motion is needed to gather the same amount of information. Moreover, as mobile users' capability to invest uninterrupted attention in a mobile device is known to be compromised (Oulasvirta et al., 2005), the interface should allow them to display what ever is currently needed as quickly and easily as possible, without complicated manoeuvring that requires all of the user's attention.

This book chapter summarises three years of research and development efforts on *m-LOMA*, a 3D mobile map of an urban environment (the city centre of Helsinki) (Nurminen, 2006). Our goal has been to create an efficient interface for navigating in a 3D view with the limited resources available. We first review general goals for designing a 3D navigation interface specifically for mobile devices. A model of user interaction with 3D maps is then presented. Subsequently, we formalise the problem of mapping controls to manoeuvring, and proceed to present mobile input devices along with those of their innate and practical problems that should be considered in the interaction design. We then present and discuss several manoeuvring schemes. The main part of the text concerns real and simulated cases and introduces a set of rules, tools, and metaphors that can ease navigation.

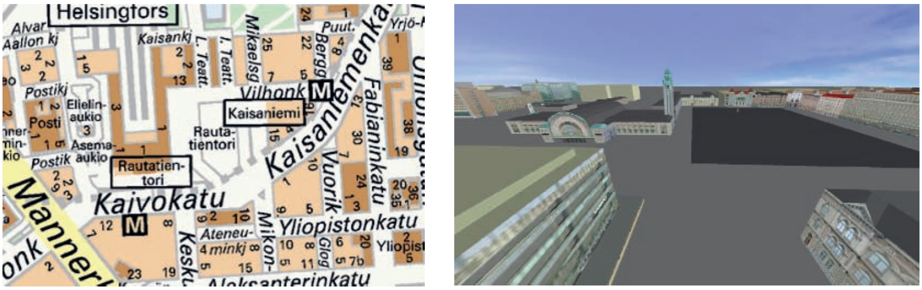


Fig. 10.1. A 2D and a 3D map of the same area.

10.2 Definitions

The term *3D map* has been used widely in the literature, assuming an intuitive definition. However, this term can be applied to representations whose source data is purely two-dimensional. Correspondingly, a data set which is in fact three-dimensional can be visualised in various fashions resembling a 2D map.

Based on our observations, we have identified two processes involved in producing 3D maps. First, a three-dimensional representation of a physical environment is formed. This representation can be a data structure, such as a 3D model, where three-dimensional shapes and visual characteristics are stored. The modelling process can simplify the real-world geometry and its visual features, but maintains the spatial relations of objects. Metadata can be included in the representation. Second, this representation is visualised as a two-dimensional image, depicting the three-dimensional aspects of the original environment and conveying the spatial relations of the physical structures of the area. The selected visualisation strategy to yield a final map may depend on context and purpose, emphasising different characteristics of the environment. We define a 3D map to be a *two-dimensional visualisation of a three-dimensional representation of a physical environment, emphasising the three-dimensional characteristics of this environment.*

3D maps may have a variety of characterstics, depending on their contents, purpose, visualisation and other features. Table 10.1 presents a common set of such attributes. In addition, there may be other run-time selectable features and modes, such as *perspective* or *orthogonal* projection, *first-person view*, or visualisation styles. For example, *thematic* visualisation could employ cartographic conventions to colour the buildings based on their type (landmark, office building, shopping mall) instead of applying textures to the façades (Plesa and Cartwright, 2007).

By our definition, a photograph is not a 3D map, as it is not created by a two-stage process, even though it is visually indistinguishable from the theoretical *ideal* map. A 2D road map with a perspective projection would not be a 3D map, as the data structures are not truly three-dimensional. Such a map could be called a "2.5D" map. 3D geometry, portrayed directly from above with orthogonal projection, could be classified as a 2D map, unless the visualisation provides clearly three-dimensional characteristics, such as shadows. Classification of maps based on procedural models would depend on the resulting model geometry and the chosen visualisation method.

Table 10.1. Common 3D map attributes.

Attribute	Explanation
Ideal	The data set and its visualisation exactly match the real world; a single image is indistinguishable from a photograph.
Realistic	Map data is visualised in a realistic manner, in an attempt to approach the ideal representation. Typically, this involves use of textures created from the real environment by digitisation.
Real-time rendered	Visualisation is performed on-the-fly instead of displaying pre-rendered animation sequences or images.
Navigable	Allows users to control the position and direction of the virtual camera.
Interactive	A navigable, real-time rendered 3D map, responding to users' queries and providing navigation-assisting features.
Dynamic	Contains time dependent elements other than the virtual camera, such as positions of GPS tracked users, public transportation etc.

Electronic	Emphasises the computerised means in producing the 3D view instead of a drawing or painting.
Urban/outdoor/indoor	Description of the represented environment.
Mobile	Electronic, running on a mobile, battery-operated device, such as a PDA or smart phone. Also implies navigability, interactivity and real-time rendering.
Immersive	A stereo projection system, where a separate view is produced for each eye to achieve an illusion of being immersed in the environment.

10.3 General requirements for mobile navigation interfaces

The design of a mobile map suited for users' needs is a challenging task, and may require several trade-offs (Meng and Reichenbacher, 2005). With mobile 3D maps, the situation is further accentuated. This section summarises user requirements for the design of mobile navigation interfaces. A real-life example of cumbersome interaction is presented.

10.3.1 Support for use in multitasking situations

When a user operating an application is completely concentrating on the task, unaware of the existence of the interface, the interface is said to be *transparent* to the user (Norman, 1988). This should be achieved also for 3D navigation. Mobile users have short attention spans, in extreme mobile situations in the order of just few seconds (Oulasvirta et al., 2005). Therefore, the mobile application should provide prompt aid. Moreover, users' operative modalities are often limited; for example, they may need to operate with one finger only, or may not hear anything due to environmental noise. In addition, users may have limited time or motivation to learn complex controls. To summarise, we assign three goals for mobile controls in pursue of transparency:

- minimise cognitive load (as defined by working memory load, amount or duration of cognitive task processing, or complexity of mental computations),
- minimise motor effort and procedural complexity,
- minimise use of time.

10.3.2 Support for navigation

While experiencing a city, the user may be performing one of the many tasks related to navigation. She may be exploring the environment, observing interesting features, searching for something, or conducting a *naïve search*, that is, extensively and at times exhaustively searching the environment (Darken and Sibert, 1996). Or, she may already know the approximate position of a target, proceeding for a closer look, on a *primed search* (Darken and Sibert, 1996). Perhaps a route has been provided, and the user is attempting to maintain orientation while manoeuvring towards the target, or is spotting the next turn point. While the user moves and observes the environment, she simultaneously develops a cognitive map (Downs and Stea, 1977). At all times, the user attempts to maintain a sense of orientation and avoid any situations that might lead to disorientation.

To support navigation, we assign the following goals for the 3D view:

- maximise information that helps orientation,
- maximise information that helps performing the current task,
- minimise information that leads to disorientation,
- maximise information that helps forming an accurate cognitive map.

Fulfilling these objectives ensures that the user

- does not get lost,
- is able to find and visit all places of interest,
- is able to re-visit places,
- feels familiar with the space.

10.3.3 Support for embodied interaction

In general, what makes mobile maps distinct and different from typical virtual environments (VEs)—such as virtual reality and desktop-based navigation systems—is that the user is *physically embedded in the world that the virtual model represents*. The dual presence is not symmetric: the roles of eye, head and body movements for acquiring information are emphasised in physical environments (PEs), whereas VEs are typically associated with decreased field of view and low fidelity of landmarks and non-visual cues. There are four implications of this.

First, what makes mobile maps different from VEs and map artefacts is the strong influence of the *keyhole property* (Woods and Watts, 1997). The keyhole property means that “the number of potential views is much greater than the physical size of the available viewports” (p. 619). Since direct recognition is often not possible because of this property unless the target happens to be on the display, users have to *move* within the space or spaces to achieve a position in which the target can be found. In contrast to VEs, mobile maps assume search within *two* spaces instead of just one.

Second, because of the keyhole property, the *alignment* of the representation with the represented space is often difficult. Alignment is important for orientation, because human spatial knowledge is known to be *viewpoint-dependent*. This also concerns knowledge of *dynamic* scenes (Garsoffky, Schwan, and Hesse, 2002). Hence, when objects in a map do not correspond to stored representations, the user has to transform or rotate the representation, which entails mental or physical effort (Levine, 1982). Mou and McNamara (2002) elaborate this view by arguing that spatial memories are defined with respect to intrinsic frames of reference, which are selected on the basis of egocentric experience and environmental cues. The availability of cues is exactly where 3D maps differ from other representations. It is also worth mentioning that viewpoint-dependence of spatial knowledge may be exaggerated in the case of mobile maps where the size of display is small. Presson, DeLange, and Hazelrigg (1989) claim that small displays do not "afford", as does movement in PE, large spatial arrays to be coded by the perceptual system during movement. Moreover, they do not support as much perceptual exploration and scanning (cf. Roskos-Ewoldsen et al., 1998).

Third, because the use of mobile maps is supposed to be possible when the user is moving, the map's support for *spatial updating*—the mechanisms involved in locating positions in space relative to oneself after a given spatial transformation—is emphasised (Wraga, 2003). Wang and Brockmole (2003) have examined what happens when the environment is divided into nested structures (e.g., city consisting of district consisting of blocks). They noted that spatial updating is not carried out in all structures simultaneously nor with the same accuracy. When switching to a new environment, one often loses track of one's position relative to old environments. Providing alternative views in 3D might support the user in updating and re-constructing such structures when needed.

Fourth, the small *scale* of transformations (e.g. Golledge, 1999) in mobile maps may lower the informativeness and recognisability of objects. Only recognisable objects can be utilised as landmarks that help the tasks of mapping and orientation. Therefore, 2D is different from 3D by trading off informativeness of an object to informativeness of an area. Witmer, Sadowski, and Finkelstein (2002) showed that adding aerial views enhances users' ability to *navigate* through VE. Furthermore, in order to help in understanding the relationship between the virtual and the physical, landmarks in the virtual model also have to be *distinctive* (stand out), which depends on the visual as well as the structural qualities of the view (Sorrows and Hirtle, 1999).

In section 10.5, several solutions to these problems are presented and analysed.

10.3.4 3D navigation with direct controls: example from a field study

To exemplify an *inefficient* search strategy, let us present a case of manoeuvring in an urban canyon with direct mapping of controls to the navigation state (See 10.5.1.). The case is taken from our field experiment (Oulasvirta, Nurminen, and Nivala submitted), at a time when the subject has completed about half of the tasks. In that experiment, the subjects were shown a target building in the virtual space and asked to move to the corresponding object in the physical space.

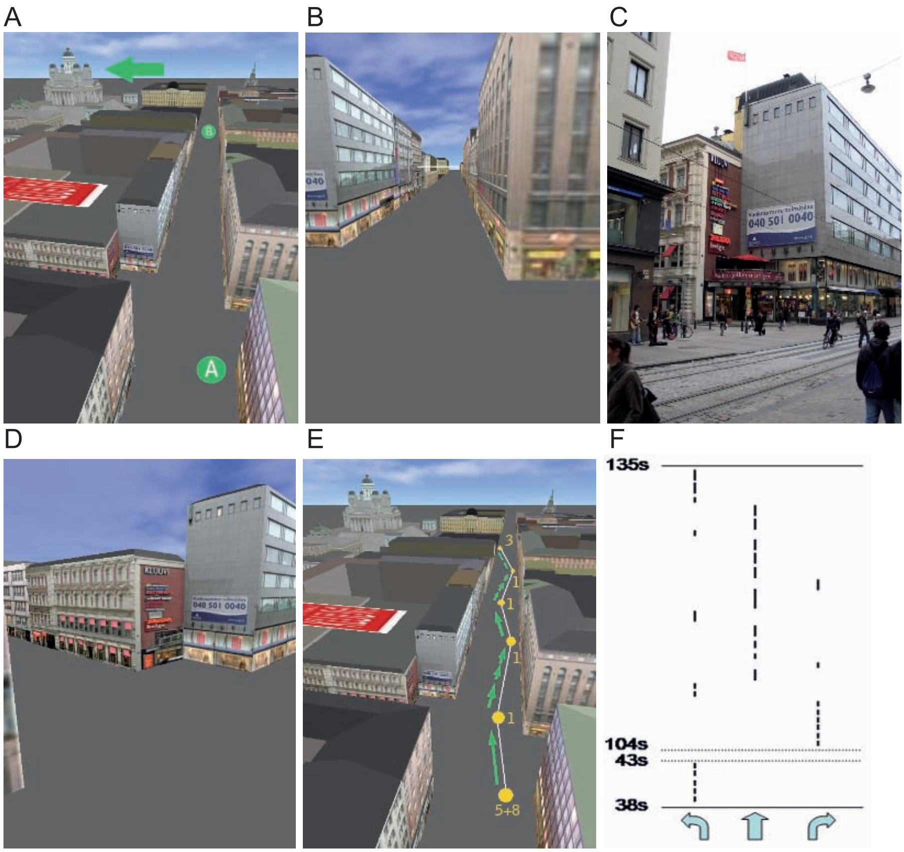


Fig. 10.2. Analysis of a real navigation episode. Explanation given in text. (Adopted from Oulasvirta, Nurminen, and Nivala submitted)

A test subject[5] located at *A* attempts to manoeuvre a 3D map view from *A* to *B* (Fig. 10.2A), to spot a landmark known to her in advance. The 3D map is running on a PDA with initial orientation of Fig. 10.2B. The map view is controlled by the PDA's joypad, which switches the viewpoint forward or backward, and rotates it left or right, as long as the control is held down. Manipulating the joypad moves the viewpoint at a constant speed and rotates it at a constant angular velocity. Additional controls affect elevation and pitch, but roll is fixed to a constant up direction. Thus, the subject has *four degrees of freedom* available for controlling the viewpoint. In the beginning of this task, the subject orients to the environment, spotting a recognisable façade (Fig. 10.2D in m-LOMA and 10.2C in the real world). The subject then determines a direction of movement and starts to manoeuvre along a road. After two blocks, the subject finds the target landmark and stops. Fig. 10.2E presents the path and points of orientation resulting from performing the task. Fig. 10.2F presents the related joypad state as

[5] The subject had trained the controls 15 mins during a training session, and about 30 mins during similar tasks at field before this one, without prior experiences of 3D games or 3D maps.

a function of time. To perform this simple task, the user performed a total of 20 rotations and 10 forward movements, using only two degrees of freedom from the available four. The subject did not move backward. The task took 135 seconds to complete, and the controls were used in two sequences: 5 seconds during initial orientation and 29 seconds during manoeuvring towards the target. The remainder of the time, the subject observed the environment and waited for a car and a tram to pass (the tram remained in view for almost a minute). All actions on controls were performed sequentially.

The presented case contains three of the four stages of navigation presented by Downs and Stea (1977): 1) the initial orientation at start, 2) manoeuvring towards a target and 3) recognising the target. The final stage, 4) maintaining the orientation to the target, is not relevant, as the distance to travel after the target has come into view is short and the route straightforward. The task appears simple, but the actual manoeuvring by the subject is unnecessarily complex. At several points, the viewpoint approaches a wall, and correcting the orientation provides only a temporary solution, as yet another correction must soon take place. Even when the subject simply rotates around one axis, a number of button presses is required to find a suitable view.

This real episode taken from our data clearly illustrates the fact that four degrees of freedom are too much for the majority of users. In section 10.5 we will go through several interface solutions that address this problem. We shall make a distinction between *manoeuvring*—actions towards a subgoal such as the orientation or moving one block, and *micro-manoeuvring*—corrective or adjusting actions that happen within a manoeuvre (such as rotating back and forth several times to find a suitable view or direction of travel).

10.4 A model of interactive search on mobile maps

Before turning to concrete design solutions to the DOF problem, we need to explain some aspects of how users interact with mobile maps in real life situations. The model of Oulasvirta, Nurminen, and Nivala (submitted) is rephrased here.

In their influential paper, Kirsh and Maglio (1994) distinguished between two kinds of *action*: *pragmatic* and *epistemic*. The former refers to action that transforms the physical problem space, for example moving a disk from one pole to another in the Tower of Hanoi task. The latter refers to action that does not directly contribute to the solving of the problem; rather, its objective is to have an *effect on the cognitive state of the agent* itself. Epistemic action can have three functions:

1. to reduce time complexity (how long something takes),
2. to reduce space complexity (how much cognitive processing is required),
3. to reduce uncertainty (how certain is an outcome) in the problem.

For example, Tetris players often quickly rotate a zoid around one or several times after it appears in an attempt to *see* how the zoid fits the landscape beneath it. This in effect changes the task from mental rotation (slow) to recognition (fast).

10.4.1 Pragmatic search action

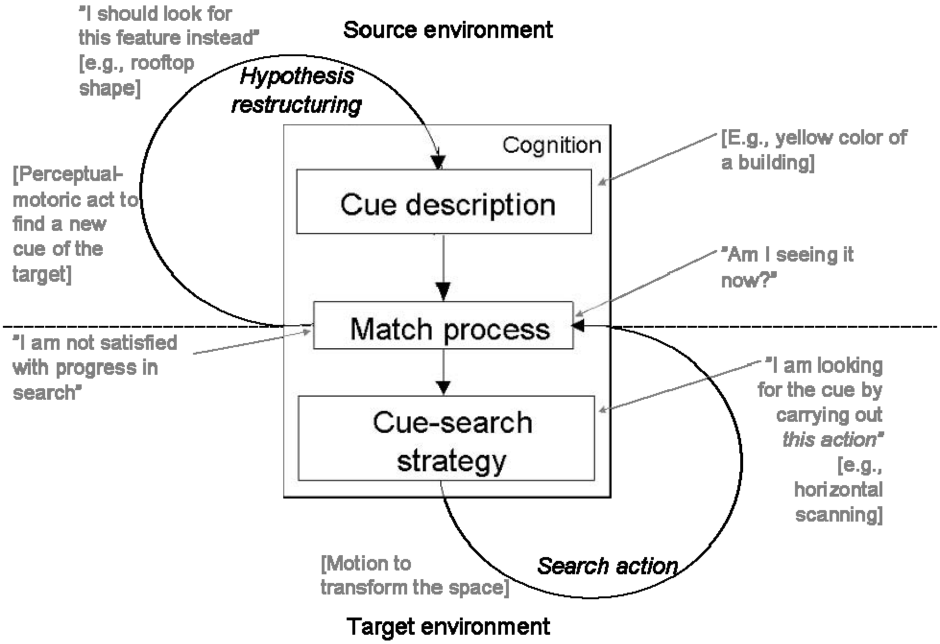


Fig. 10.3. A model of interactive search with mobile maps as pragmatic action

Fig. 10.3 presents *the model of pragmatic search* (cf. Jul and Furnas, 1997). In the core of the model are 1) a match process taking care of the comparison between the target description (kept in mind) and the perceived space and 2) a motor action process that transforms the space according to the cue-search strategy. The *search-action loop* involves acting based on a selected strategy for searching for a cue (e.g., searching for a certain rooftop shape by scanning horizontally while keeping view toward rooftops). Carrying out the strategy produces a change in the state of the world. This new state is perceived and a new match is attempted. A failed matching process leads back into the search-action loop. Importantly, search action can take place both in the virtual and the physical world, unlike in VEs where the search always takes place in the virtual space. When the current strategy does not produce satisfactory matches, it has to be changed. This triggers the *hypothesis-restructuring loop*, which involves acquiring a new target description by choosing a new cue in the source environment (e.g., noticing that the target building is the lowest one). This does not always happen as an internal process, but is often manifested in the subject physically returning to a position where a new description of the target can be extracted or the earlier one elaborated.

10.4.2 Epistemic search action

Below, we report a total of seven behaviours, observed by Oulasvirta, Nurminen, and Nivala (submitted), that can be interpreted as *epistemic action* rather than pragmatic action (for visualisations of related search paths, see Fig. 10.4):

1. Scanning the immediate context of the target, in order to elaborate the description of the target held in working memory (e.g., by looking at colours of neighbouring buildings).
2. Scanning for landmarks such as statues, in order to elaborate the description of the target or to use prior semantic and spatial information.
3. Egocentric positioning (locating oneself in the map), in order to utilize the (stronger) representations of the PE in action for the VE.
4. Circling (walking around a full circle in VE) and viewing the buildings, thus creating initial mental representations to support search later on.
5. Exploring the proximal area (e.g., moving around in an area of few blocks from the starting position) in the first experimental trial to familiarise oneself with the model, thus reducing search costs in the subsequent trials.
6. Peeking around a corner to identify action alternatives and familiarise oneself with the surroundings, thus elaborating a representation of the vicinity and reducing uncertainty in the upcoming trial.
7. Rotating: the view is quickly rotated horizontally in the beginning of the trial to see the surrounding buildings. May serve egocentric positioning, landmark search etc. (Similar to tactics 4, 5, and 6.)

But what is actually achieved by such epistemic actions from the perspective of the task of finding a given target? We posit that there are three main *functions* of epistemic actions in interacting with a mobile map:

1. *Improving cue descriptions.* In some of the epistemic actions, the user scans the target and its immediate neighbourhood. This can result in elaboration of the mental representation of the target, which in turn can facilitate the subsequent match process (i.e., reduction of uncertainty).
2. *Improving match process.* Some of the epistemic strategies can be explained as attempts to shift the processing burden from the limited and effortful short-term working memory system by switching from mental computation and maintenance to *recognition.* This strategy reduces uncertainty and decreases the time complexity of the task.
3. *Improving search strategy and search action.* Naturally, richer representations of the environment can participate in strategy selection and implementation of search action. For example, having a hypothesis of the target's position in relation to a landmark helps decisions on where to look. This can make exhaustive scanning unnecessary. Such epistemic actions as exploring and peeking in effect enhance the user's possibility to know where to search for the target by narrowing down the number of alternatives.

Implications of this descriptive user model to design are discussed in section 10.5.

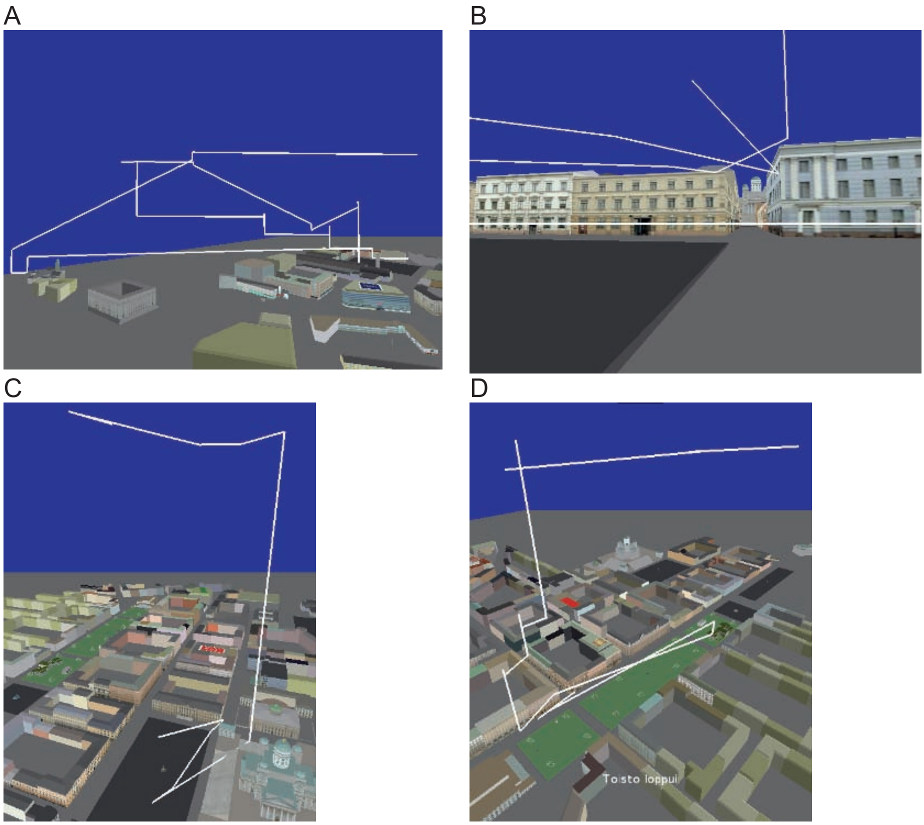


Fig. 10.4. Visualisations of users' epistemic actions when searching a target. A) Learning the model by exploring it from a top-down view; B) Walking around in a city square and peeking around corners; C) Diving to street-level; D) Scanning, diving and scanning at multiple levels, and walking around at the street level. From the movement log data of the experiment of Oulasvirta, Nurminen, and Nivala (submitted).

10.5 Designing controls

An integral part of interaction design of navigation is the assignment of interface controls to movement and action in the mobile map environment. Typical mobile devices capable of running 3D graphics include personal digital assistants (PDAs) and smart phones (see Fig. 10.5). The controls offered by PDAs commonly include a touch screen, a few buttons, and a joypad. The touch screen essentially provides a direct pointing mechanism on a 2D plane, and a two-dimensional analog input. The touch screen is operated by a stylus (a pointing stick). The buttons and the joypad are simple discrete inputs. A direct text input method may be missing due to the lack of buttons, but in that case, it is compensated by a virtual keyboard, operable by the stylus. A smart phone commonly provides only discrete buttons. Usually a few of the buttons are assigned to serve menu selections, and the rest are used either in typing text or dialling. Sometimes a joypad is provided for easier menu navigation. Exceptions to theses prototypical examples exist, such as Nokia's Communicators, which include a full keyboard. Some even contain both a keyboard and a touch screen, such as the Palm Treo 650. Sometimes, a small keyboard can be attached to a PDA.

When a control state is propagated to, and received by an application, it is called an *input event*. Mobile operating systems commonly prevent input events to be generated in parallel, in a forced attempt to reduce possible control DOFs to 1; if one button is down, the next possible event is the up event from that button. Platform-dependent software development kits provide information to override this restriction.

Fig.10.5. Mobile devices differ in the availability and layout of interfaces.

10.5.1 Mapping controls to navigation

The use of controls can be described as a *mapping* from control space G to navigation space N. The navigation space can be described by a navigation state, which contains all the variables related to navigation, including the camera position and orientation, but also other variables such as speed, field of view (FOV) etc. The mapping can also be called the function g of the input i, g: $G \rightarrow N$. We will call the number of relevant user actions the *control DOF* and *the navigation DOF* the number of variables available for movement. The mapping provides movement on a guide manifold, a constrained subspace of the navigation space (Hanson and Wernert, 1997). Discrete control inputs can be used for providing either a single event or multiple independent discrete events. The mapping can depend on time, possibly in a nonlinear manner. A 2D input can be used similarly for a single 2D event, a discrete sequence of such events $(x, y)_j$, or a time dependent input (x,y,t). The motion derivatives (x',y') can also be used as an input. If a control is given cyclic behaviour, the mappings depend on the current cycle c of the inputs, g: $G_{ci} \rightarrow N$. In this case, one button can control two or more functions, each button release advancing the function to next on the cycle.

10.5.2 Control delays

Physical controls provide a multitude of error sources. A single binary discrete event may simply not occur when a button malfunctions. Time-dependent events are prone to errors in timing. This can be caused either by the user or by the device. The user simply may not be able to estimate timings accurately, but the same may be true for the device. For example, with the Symbian operating system, the accuracy of the system clock currently available to applications is less than 20ms (1/64s). There might be constant minimum delays between events, caused by an operating system or a device controller, or the controller output may be sampled at discrete intervals, causing seemingly random output timings if the sampling rate is too low. Event propagation from the OS to the application may also depend on available CPU, which may be sparse during 3D rendering. For example, when a user presses a rotate button, rendering may start immediately, but when the button is released, it can take a while before the rendering engine receives the event. If the user was attempting to aim at a certain location, the possible crosshair would have passed the target. Combining these problems may yield a temporal, context-dependent inaccuracy of worse than 100ms. While a coherent delay can be anticipated by a user, incoherent temporal errors are difficult to adapt to. Moreover, all such delays contribute to degradation of user experience and, consequently, of usability. Therefore, identifying least-latency control mechanisms is among the very first tasks of 3D interaction design for mobile devices.

10.6 Designing for navigation

Navigation is a process involving both mental and physical action, both way-finding and movement (Darken and Sibert, 1996). All movement requires manoeuvring, performing a series of operations to achieve subgoals. Whereas manoeuvring in a 2D view can be mapped to a few input controls in a relatively straightforward manner, movement in a 3D world cannot. In addition to 3D position, one needs to specify 3D orientation as well. Direct control over such a view would require simultaneous specification of at least six degrees of freedom. Generally, producing decent motion requires even more variables, but a mobile device only has a few controls, of which the user might want to use only one at a time. We assess that developing interaction for a mobile 3D map depends heavily on solving this problem.

Therefore, when designing mobile maps, designers have to implement both cartographic (amount of information presented, symbolisation, generalisation, simplification) and interaction solutions, and this latter part is often overlooked. The model of pragmatic action presented above suggests that the key way to minimise the possibility of choosing a failing strategy is to guide and direct the user to use those cues that are known to be efficient in that particular model (e.g., for m-LOMA, façades of buildings are more effective than street geometry). Similarly, designers can deliberately make it more difficult to use those cues that are inefficient. Here, This design strategy is called *guidance*.

Second, the model of epistemic action suggests supporting 1) the elaboration of the target description, 2) the construction of a representation of the area in the PE, 3) the use of prior knowledge, and 4) transforming the locus of task processing from working memory to perceptual modules. However, if guidance was optimally effective, one could argue that users would not need to relapse to epistemic action and other "corrective" behaviours. This, we believe, is not the case. Because of substantial individual differences in representing the environment and in the use of cues and landmarks (e.g., Waller, 1999), and because information needs vary between situations, the best solutions are those that support *flexible switches* between efficient strategies.

Manoeuvring in a VE can be realised with various levels of control over movement. Table 10.2 presents a set of manoeuvring classes, in decreasing order of navigation freedom. Beyond simply mapping controls to explicit manoeuvring, one can apply metaphors in order to create higher-level interaction schemes. Research on virtual environments has provided several metaphors (see Stuart, 1996). Many but not all of them are applicable to mobile 3D maps, partly due to restrictions of the input methods and partly due to the limited capacities of the user. Several methods exist for assisting or constraining manoeuvring, for guiding the user's attention, or for offloading unnecessary micro-manoeuvring. For certain situations, pre-animated navigation sequences can be launched via shortcuts. With external navigation technologies, manoeuvring can be completely automatic. It is essential that the special circumstances and potential error sources typical to mobile maps are taken into consideration in navigation design. Selecting a navigation scheme or metaphor may also involve striking a balance between support for direct search for the target (pragmatic action) on the one hand and updating cognitive maps of the area (epistemic action) on the other. In what follows, several designs are presented, analysed, and elaborated in the framework of navigation stages (Downs and Stea, 1977) from the user's perspective.

Table 10.2. Manoeuvring classes in decreasing order of navigation freedom.

Manoeuvring class	Freedom of control
Explicit	The user controls motion with a mapping depending on the current navigation metaphor.
Assisted	The navigation system provides automatic supporting movement and orientation triggered by features of the environment, current navigation mode, and context.
Constrained	The navigation space is restricted and cannot span the entire 3D space of the virtual environment.
Scripted	Animated view transition is triggered by user interaction, depending on environment, current navigation mode, and context.
Automatic	Movement is driven by external inputs, such as a GPS device or electronic compass.

10.6.1 Orientation and landmarks

The first stage of any navigation task is initial orientation. At this stage, the user does not necessarily possess any prior information of the environment, and her current position becomes the first anchor in her cognitive map. To match this physical position with a 3D map view, external information may be necessary. If a GPS device is available, the viewpoint can be commanded to move to this position. If the map program contains a set of common start points potentially known to the user, such as railway stations or major bus stops, a selection can be made from a menu. With a street database, the user can walk to the nearest intersection and enter the corresponding street names. When the exact position is known, the viewpoint can be set to the current position, perhaps at street level for a first-person view. After resolving the initial position, we further encourage assigning a visual marker, for example an arrow, to point towards the start point. If the user's attempts at localisation fail, she can still perform an exhaustive search in the 3D map to find cues that match her current view in physical world.

For orientation purposes, landmarks are essential in establishing key locations in an environment (Evans, 1980; Lynch, 1960; Vinson, 1999). Landmarks are usually considered to be objects that have distinguishable features and a high contrast against other objects in the environment. They are often visible from long distances, sometimes allowing maintenance of orientation throughout entire navigation episodes. These properties make them useful for epistemic actions like those described in section 10.4. To facilitate a simple perceptual match process, a 3D map should reproduce landmarks in a directly recognisable manner. In addition, a 3D engine should be able to render them from very far distances to allow visual searches over entire cities and to anchor large scale spatial relations.

Given a situation where the start point has been discovered, or the user has located landmarks in the 3D map that are visible to her in PE, the user still needs to match the two worlds to each other. With two or more landmarks visible, or a landmark and local cues, the user can perform a mental transformation between the map and the environment, and triangulate her position (Levine, Marchon and Hanley, 1984). Locating landmarks on a 3D map may require excessive micro-manoeuvring, even if they are visible from the physical viewpoint. As resolving the initial orientation is of such importance, we suggest assigning a direct functionality to it. The *landmark view* would automatically orient the view towards landmarks or cues as an animated view transition, with one triggering control (a virtual or real button, or a menu entry). If the current position is known, for example with GPS, the landmark view should present both the landmark and the position. Without knowledge of the current position, the same control would successively move the camera to a position where the next landmark is visible. Implementation of such functionality would require annotating the 3D model with landmark information.

Sometimes, no major landmarks are visible or in the vicinity. In this case, other cues must be used for matching the virtual and real environments, such as edges or areas, street names, topological properties, building façades, etc. Local cues can be unique and clearly distinguishable, such as statues. Some local cues, such as restaurant logos, are easy to spot in the environment even though they are not unique. We suggest populating the 3D environment with local cues, *minor landmarks*, and providing the system with related annotation information. Again, a single control would trigger camera animation to view the local cues. As this functionality draws the attention of the user to local cues, it requires knowledge of the user's approximate position to be effective.

As landmarks are often large objects, we suggest assigning landmark annotation to entire entities, not only to single points. An efficient 3D engine with visibility information available can enhance the landmark view functionality by prioritising those landmarks that are at least partially visible to the user in PE.

10.6.2 Manoeuvring and exploring

After initial orientation is obtained, the user can proceed with any navigational task, such as a primed search (Darken and Sibert, 1996). In a primed search, the target's approximate position is resolved in advance: a point of interest could be selected from a menu, the user could know the address and make a query for coordinates, a content database could be searched for keywords, or the user could have a general idea of the location or direction based on her cognitive map. A primed search consists of the second and the last of navigational stages, that is, manoeuvring close to the target and recognising the target during a local browse. We suggest assigning another marker arrow to the target.

The simplest form of navigation would be immediately teleporting the viewpoint to the destination. Unfortunately, instant travel is known to cause disorientation (Bowman et al., 1997). The commonly suggested way of travelling to long distances in generally straightforward direction is *the steering metaphor*, where the camera moves at constant speed, or is controlled by accelerations. By controlling the acceleration, the user can define a suitable speed, but doesn't need to use the controls to maintain it, relieving motor resources for orientation. Orientation could indeed be more directly controlled while steering, in order to observe the environment. In an urban environment, moving forward in a straight line would involve positioning the viewpoint above rooftops in order to avoid entering buildings.

If the user is not yet willing to travel to a destination, she could start exploring the environment as epistemic action, to familiarise herself with it. Again, controls could be assigned according to the steering metaphor. For a better overall view of the environment, the user should be allowed to elevate the virtual camera to a top-down view, requiring an additional control to turn the view towards the ground. This view would allow her to observe the spatial relationships of the environment in a metrically accurate manner. If the user wishes to become acquainted with the target area without unnecessary manoeuvring, the *click-and-fly* paradigm can be applied, where the user selects a target, and an animated view transition takes her there. Animated view transitions should also be possible when start and end points are defined, for instance by selecting them from a list of known destinations or by having direct shortcuts assigned to them.

10.6.3 Maintaining orientation

When a user is navigating in an environment, during exploration or on a primed search towards a target, she should constantly observe the environment to enrich her cognitive map. Frequent observations are necessary for maintaining orientation, and learning the environment decreases the user's dependency of artificial navigational aids. Where major landmarks provide a frame of reference, local (minor) landmarks help making route decisions (Steck and Mallot, 1998).

Following the work of Hanson and Wernert (1997), we suggest using *interest fields* as a subtle approach to drawing the user's attention to cues in the environment. When the user manoeuvres in an environment, an *assisted camera* scheme points the camera towards landmarks or local cues such as statues or restaurants with noticeable logos. The attentive camera metaphor (Hughes and Lewis, 2000) suits this automatic orientation well. It orients the view towards interesting cues, but lets the movement continue in the original direction. When the angular distance between movement vector and view vector becomes large, the view returns to pointing forward. In addition, the assisted camera could support orientation (Buchholz, Bohnet, and Döllner, 2005; Kiss and Nijholt, 2003). When the camera is elevated, this scheme automatically orients the camera slightly downwards, in order to avoid filling the view with sky. The user can intervene in the suggested assistance and prevent it with a single click on a control opposite the orientation direction.

In cases where distinguishable local cues are missing, the local position and orientation can be verified directly with features that have been included in the 3D model, such as building façades. Individually textured façades provide a simple way of matching PE and VE almost anywhere. Unfortunately, not all façades provide distinguishable features (or are otherwise memorable), to which end the guidance provided by the system should prioritise other cues, if present.

During the initial orientation, the user was provided with a button that triggers a scripted action for viewing the closest landmark. When she is manoeuvring, the interest fields will mainly be guiding her attention to new local cues, or she can verify her position from other features such as building façades. However, such local information will not necessarily develop her cognitive map, and neglecting to frequently observe known anchor positions can lead to disorientation. Therefore, it is advisable to reorient the view to known landmarks from time to time. The user can achieve this using the same landmark view operation that was used initially, showing one or more landmarks, and then returning to normal navigation mode. Or, the system can suggest this action automatically, as an assisting feature.

An example of the assisted camera scheme is provided in Fig. 10.6A-6D. When the user first approaches a landmark, the system provides the view presented in Fig. 10.6A (at the user’s discretion). The user’s current position is marked with a red dot. Fig. 10.6B presents the user’s path, depicted with a long arrow. As the user approaches a corner, the view is automatically oriented towards the landmark (10.6C), and returned to normal view as the user proceeds forward. After a while, the system suggests looking backward (Fig. 10.6D). In Fig. 10.6A, note the two other landmarks in the horizon. Fig. 10.6D includes two local cues, a statue and a bar’s logo. Automatic orientation in such a manner requires optimisation of the view's orientation value based not only on elevation, but the presence of visible cues and landmarks.

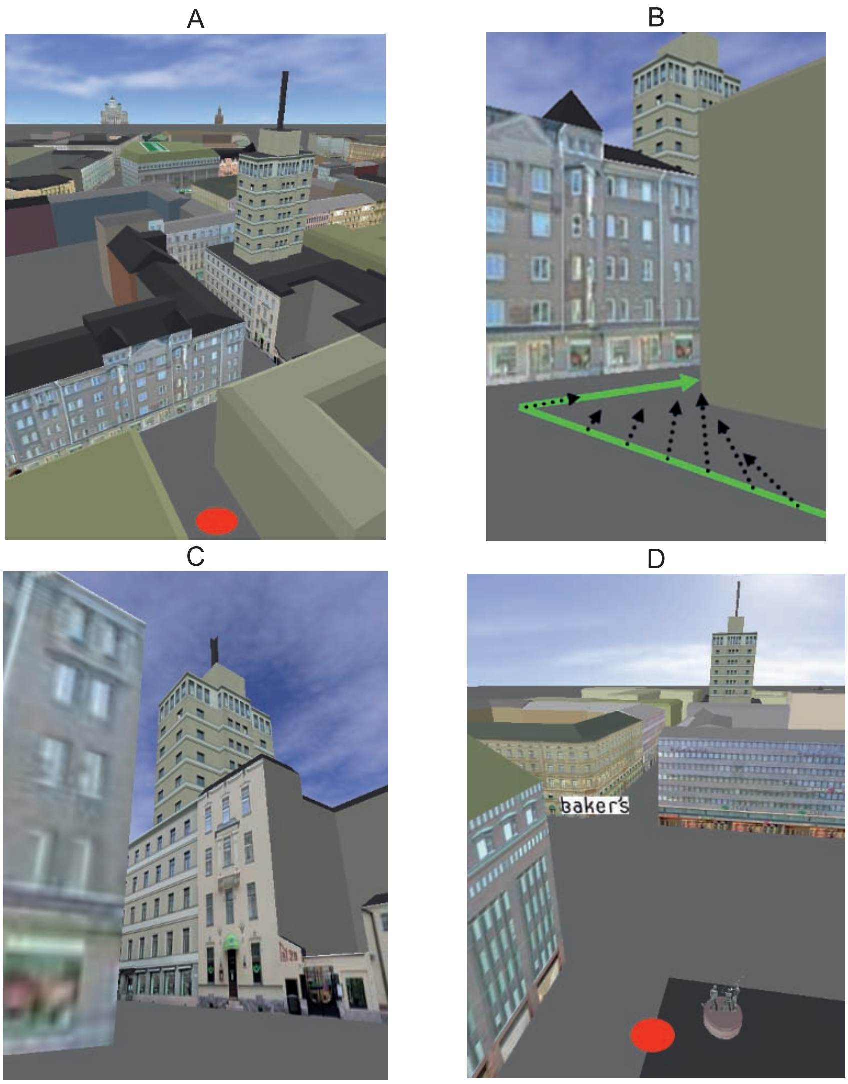


Fig. 10.6. An assisted camera scheme. When approaching a landmark (the tower), a quick overall view (A) is suggested. As the landmark comes into view, an automatic glimpse is provided (B and C). When the landmark has been passed, an overall view is suggested again (D).

10.6.4 Constrained manoeuvring

Manoeuvring above rooftops appears to provide a simple, unconstrained 3D navigation space. However, one of the strengths of a 3D map is the possibility of providing a first person view at street level. Unfortunately, manoeuvring at that level will immediately lead to the problem of entering buildings through their façades, which is known to cause disorientation. The solution is a collision avoidance scheme that keeps the viewpoint outside objects. The simplest form of collision avoidance merely prevents movement when a potential collision is detected, which causes micro-manoeuvring as the user must correct her position and orientation before continuing. A better solution would be to allow movement along a colliding surface, but even then the view would be filled by the façade, again causing disorientation (Smith and Marsh, 2004).

We suggest applying street topology in order to limit the navigation space. Given a street vector database that contains street centrelines, and matching the coordinate system with the 3D model, the view is forced to remain along the street vectors, staying at a distance from building façades. We will call this manoeuvring scheme the *tracks mode*. Manoeuvring in this mode consists of moving along tracks and selecting from available tracks at crossings.

The usual assisted camera scheme keeps the camera pointed towards local cues. In addition, when the user orients towards façades, the assisted camera maximises the information value by moving the camera away from that surface, inside the building behind if necessary (Fig. 10.7). The 3D engine should allow such motion, and avoid rendering the inner façade of the penetrated wall. Alternatively, the field-of-view can be widened, but that may lead to unwanted perspective distortions, depending on the situation.

10.6.5 Reaching a destination

At the end of a primed search, the user needs to pinpoint the exact goal of the search. This may require naïve search within the vicinity of the target. It may be sufficient to perform this search in the PE, but the user might also conduct it as epistemic action in the 3D map before arriving at the location. The search can be performed using the above-mentioned manoeuvring methods, perhaps at street level. Alternatively, the user can select a pivot point, around which the search is performed in a *target-oriented* manner. In this case, the navigation subspace is cylindrical and the view centred on a pivot point. An explicit manoeuvring scheme in a cylindrical navigation space would require 3 DOFs, namely radius, rotation, and elevation. A similar spherical control mapping would involve radius and angular location on the sphere surface.

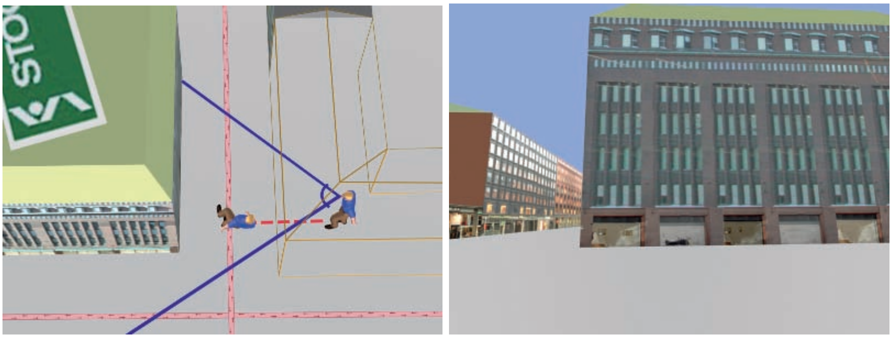


Fig. 10.7. Virtual rails keep the user in the middle of the street (left). When rotating, the distance to the opposing façade is adjusted (left) in order to provide a better view (right)

10.6.6 Complementary views

The previous sections provide cases where viewpoint is sometimes set at street level, sometimes at rooftop level, and sometimes in the sky looking down. These viewpoints are informationally complementary, each associated with different interaction modes designed particularly for finding those cues that are informative in that view. We suggest two alternatives: as already mentioned, the explicit manoeuvring scheme would include controls for elevation and pitch, which would be aided by the assistance scheme that maximises the orientation value of the view, orienting the view downwards as the elevation increases. As a second alternative, we suggest assigning a control that triggers an animated view transition between a street level (small scale: first-person view), rooftop level (medium scale: local cues visible) and top-down view (large scale: spatial relations). Assigned to a single control, this would be a cyclic action. With two controls, the direction of animation can be selected. Fig. 10.8 presents a rooftop view and a top-down view. In addition, separate 2D map views would be useful, for example to better convey the street topology. Rakkolainen and Vainio (2001) even suggest simultaneous use of 2D and 3D maps.

10.6.7 Routing

Given a topological street database, routing functionality can be implemented for example using the A* search algorithm (Hart et al., 1968). When start and end points are set, a route along the streets can be calculated and visualised. Fig. 10.8 presents a route with start and end points marked by arrows and the route visualised as a semi-transparent wall.

Routing offloads parts of the way-finding process of the user, letting her concentrate on the local cues necessary for following the pre-calculated path. While the user still could navigate freely, following a route naturally suits our constrained manoeuvring scheme. Given a route, the path is now essentially one-dimensional, and requires very little interaction from the user. With a GPS device, movement along the route would be automatic. An assisted camera scheme would constantly provide glimpses at local cues, minimising the need to orient the view. At each crossing, the assisted camera scheme would orient the view towards the correct direction.

As support for epistemic action, a separate control could be assigned to launch a walkthrough of the route, in order for the user to familiarise herself with local cues related to important decision points such as crossings.

During navigation, the user would mostly be involved in simple recognition processes, observing cues of the local environment. Our primary suggestion is to offer a street-level view, minimising the need for spatial transformations. Secondarily, route navigation could be target-oriented, the viewpoint orbiting at rooftop level around a pivot point. In this case, controls would affect the movement of the pivot point and the supposed current location. A GPS could control the position of the pivot point automatically. To maintain orientation, the user should be encouraged to keep observing large scale features such as landmarks as well, as suggested in the previous section.

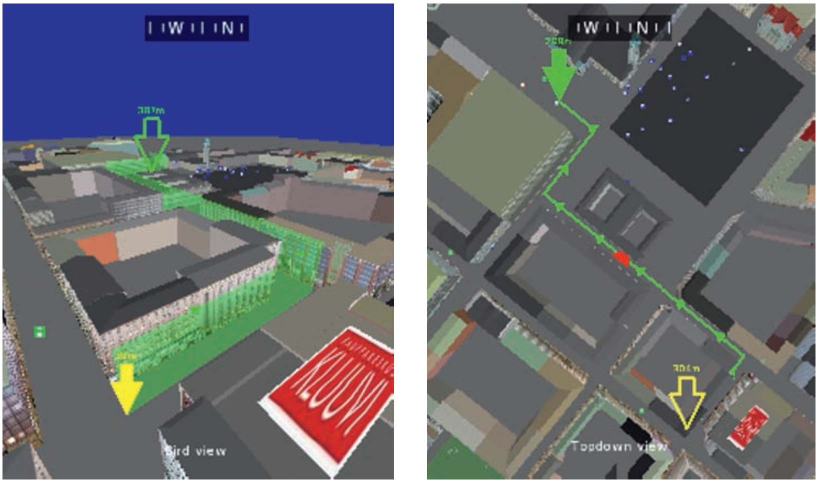


Fig. 10.8. Route guiding mode. Route visualisation in bird's eye and top-down views.

10.6.8 Visual aids

The examples above have presented a few artificial visual aids for navigation in addition to a realistic 3D model: marker arrows, a GPS position point, and route visualisation. The markers could also display the distance and the name or logo of the target. We also suggest further visual cues: for example, the arrows in our system are solid when the assigned point is visible and outlined when it is not (Fig. 10.8). In addition to the assisted camera scheme, temporary markers could be assigned to cues that lie too far away from the orientation of the view provided by the attentive camera, with transparency depicting the angular distance. When users encounter subjectively salient cues, they should be allowed to mark them as landmarks, and assign a marker as a spatial bookmark.

As overlay information, the current manoeuvring metaphor, camera assistance status, or street address could be rendered on the display. A graphical compass could also help in orientation. Fig. 10.8 presents markers with distance, a compass and current navigation mode (the most recent setting). In addition, location-based content could be integrated into the system, represented for example by billboards. If these billboards were to present graphical company logos in easily recognisable manner, they could be used as local cues for the assisted camera scheme.

10.7 Input mechanisms

In the previous section we implicitly assumed that all interaction except for animated view transitions would involve time-dependent, explicit manoeuvring. As long as a button is being pressed, it will affect the related navigation variables. We now present two alternate mechanisms to complete the interaction palette, and proceed to design an integrated navigation solution.

10.7.1 Discrete manoeuvring

With explicit, continuous manoeuvring, the user is constantly involved with the controls. The requirement to navigate both in the PE and the VE at the same time may be excessively straining, especially with an unrestricted, unassisted navigation scheme as described in section 10.3. Especially at street level, each intersection poses a challenge, as the user must stop at the correct position and orient herself accurately towards the next road before proceeding. The *tracks* mode helps by constraining the navigation space, but the user still needs to constantly manage the controls in order to manoeuvre the camera. In the case of route following, the essentially one-dimensional route may suffice, as the user mainly just proceeds forward.

As an alternative to continuous manoeuvring, discrete navigation can provide short animated transitions between positions, requiring user attention only at certain intervals. Step sizes can be configured. At crossings, angular discretisation can depend on the directions of the streets. A simple angular discretisation scheme is presented in Fig. 10.9, where rotation of the view will continue until it is aligned with one of the preset directions. The need for accuracy is reduced as the system is pre-configured. The user may be able to foresee what actions will soon be required, for example when approaching a crossing. Therefore, the system should cache the user's commands and execute them in order.

The downside of discrete manoeuvring is the lack of freedom to explicitly define position and orientation, which may reduce the possibility to observe cues in the environment. Thus, the importance of an assisted camera scheme is emphasised, as without automatic orientation towards cues, the user might not notice them.

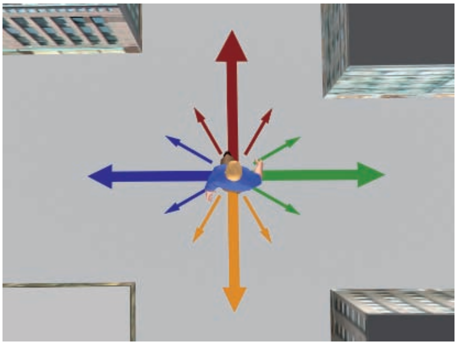


Fig. 10.9. Possible viewing and movement directions in a crossing with discrete manoeuvring

10.7.2 Impulse drive

A compromise between explicit, continuous manoeuvring and explicit, discrete manoeuvring would be *floating*, similar to steering, where controls would give the virtual camera *impulses*. Each impulse would increase the first derivative of a navigation variable, such as speed of movement or rotation. Continuous *thrust* would provide a constant second derivative, such as acceleration. Both the impulse and thrust should be configurable by the user. By setting the thrust to zero, acceleration would still be possible with a series of impulses. In all cases, a single impulse opposite the direction of motion would stop the movement. In addition, *friction* would act as a small negative second derivative (deceleration) to all navigation variables, preventing infinite movement.

10.7.3 2D controls

Several mobile devices include a touch screen, operated by a stylus. As an input device, a touch screen produces 2D position events. A single event can be used to operate software UI components, or as a direct pointing paradigm. A series of events could be produced by pressing and moving the stylus on the display. Such a control could drive navigation variables in a seemingly analogous manner, given that the events are consistent and sufficiently frequent (see section 10.5.2).

10.8 Navigation interface

Navigation in a 3D space with limited controls is a challenging optimisation task for the interface designer. The previous sections have introduced a set of navigation tasks and cases, with several supporting navigation designs and mechanisms. A real application must strike a balance between these solutions to yield a complete, integrated navigation interface.

10.8.1 Combined navigation functions

Table 10.3 presents a collection of the discussed functions and provides a selection method for each function. Shortcuts are offered only to functions that are needed relatively often. Certain functions should be allowed to affect each other. For example, if a route is defined and tracks are turned on, movement is limited to the route. Also, we turn off collision detection in orbiting mode. Available combinations are also affected by current modes. If the viewpoint is tied to the GPS, steering or floating are not available, but orbiting and selection of the level of view (street level view, bird's eye view or top-down view) are possible.

10.8.2 Control mappings

Mapping manoeuvring methods to controls depends on the available inputs. Fig. 10.10A through C present sample mappings for common PDA hardware buttons for direct movement, steering, and orbiting. Bindings and shortcuts for a touch screen are presented in Fig. 10.10D. We reserve the lower part of the screen for a menu and shortcuts. The icons from the left present shortcuts to *help, landmark view, routing widget, direct/orbit mode, fly to GPS, view transition, tracks mode* and *2D map.* Touch screen margins are mapped to *pitch* (left), *elevation* (right), *pan* (low) and *zoom* (up) in direct manoeuvring mode. Stylus movement in the centre of the screen in direct mode moves the viewpoint forward, backward or rotates it. Movement or rotation continues if the stylus reaches any of the margin areas. As a touch screen allows direct pointing, we have also implemented context-sensitive menus (Fig. 10.11). Using the *fly to* functionality, the user can perform a *point-and-fly* scripted action. The menus allow, among other things, insertion of start and end points for routing and triggering the scripted action *fly along route* (the epistemic action of an assisted walkthrough). Currently, PDA hardware buttons are assigned to discrete movement, as the touch screen provides an analog interface.

Mappings for a smart phone are presented in Fig. 10.12. Currently, all controls are assigned to explicit manoeuvring. Other functions are only available via a menu, launched by a hardware button. In smart phones, movement is currently set to be continuous for explicit manoeuvring, and discrete for the *tracks mode.*

The presented mappings are provided as en example from our implementation of a 3D map. It is advisable to let users configure the bindings to their liking, for example via a configuration file.

Table 10.3. Navigation functions.

Navigation type	Function/mode	Selection method	Comment
Explicit	Direct/steering/orbiting	Shortcut/menu	If following route, orbit around route points.
Explicit	Discrete	Menu	N/A for floating; configure impulse and thrust.
Assisted	Assisted camera	Menu	Assistance intervention possible via an action against assisted motion.
Constrained	Tracks	Shortcut/menu	Triggers animated transition to nearest road, or to route, if defined. If route defined, ties viewpoint to route.
Constrained	Route definition	Widget Point-and-define	When start and end points defined, always generate a route
Constrained	Collision detection	Menu	Assisted camera may temporarily turn off. Off in orbiting mode
Scripted	Landmark view	Shortcut/menu	
Scripted	View mode up	Shortcut/menu	Street/bird/top-down view
Scripted	View mode down	Shortcut/menu	Street/bird/top-down view
Scripted	Fly to start	Shortcut/menu/point-and-fly	If start point defined
Scripted	Fly to end	Shortcut/menu/point-and-fly	If end point defined
Scripted	Route walkthrough	Widget/shortcut/menu	If route defined
Scripted	Fly to GPS	Shortcut/menu	If GPS enabled Ties viewpoint to GPS (street/bird/top-down and orbiting applicable).
Scripted	Fly to …	Menu: POI selection Widget: address Widget: coordinates Point-and-fly	
Automatic	GPS	Menu: enable GPS	Triggers fly to GPS and bird view. Enables GPS tag and assigns a marker.

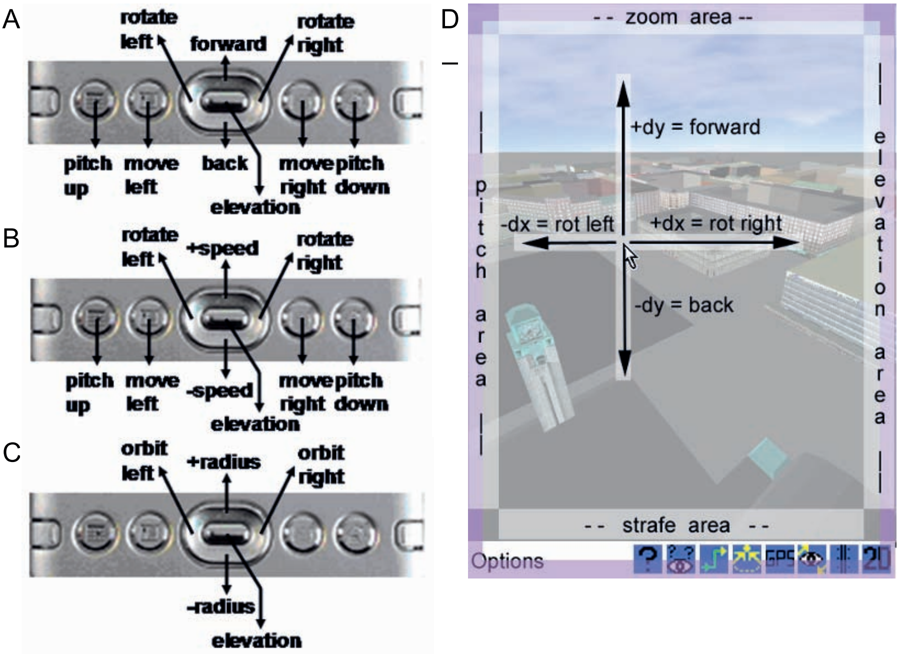


Fig. 10.10. Current controls in the PDA version of m-loma for A) direct movement, B) steering movement, and C) target-oriented movement, and D) active areas for stylus input.

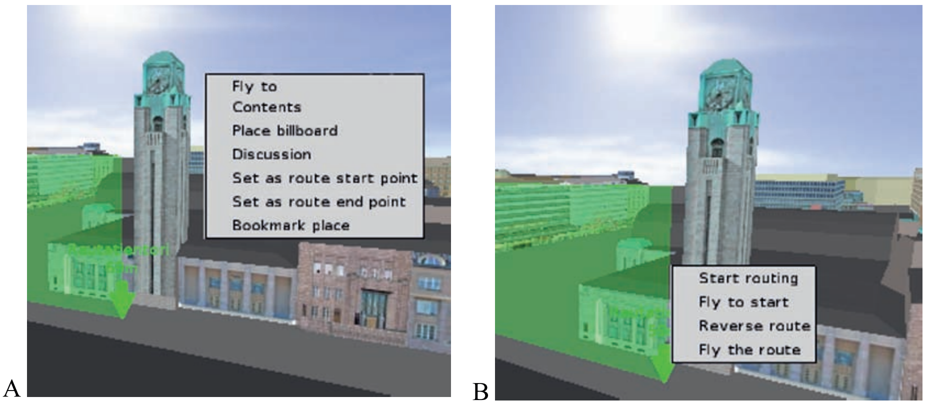


Fig. 10.11. Context menus for A) a building and B) for a route marker arrow

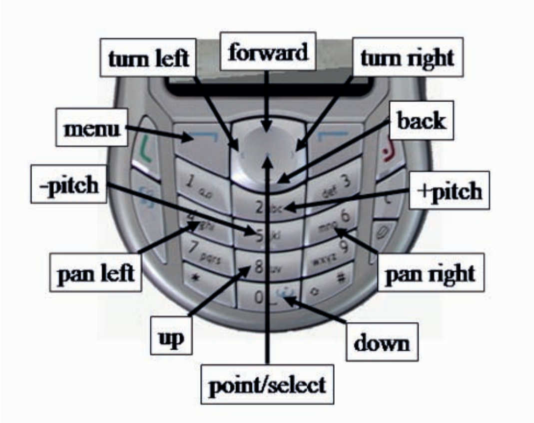


Fig. 10.12. Explicit, direct manoeuvring controls for a smart phone

10.9 Implementation notes

Several of the presented techniques require efficient implementation in order to be affordable. For example, straightforward implementation of collision avoidance may require substantial computational resources not available in mobile devices. In addition, certain functionalities depend on content management along with support from the 3D map engine. For example, landmark positions and possibly even their geometry may need to be known to the system. In order to function according to expectations, the assisted camera scheme requires visibility information, which may not be available without implementing highly sophisticated solutions. Real-time rendering of large, richly textured 3D models on mobile devices is itself a substantial technical challenge. Nurminen (2006) provides technical details on the m-LOMA system implementation.

10.10 Summary

3D maps provide several potential improvements over their 2D counterparts. Orientation can be performed visually by direct comparison between the map and the environment. During navigation, focus can be shifted from labels (street names) to direct visual cues. The success of this shift depends on the design of the cues and the user interface. Nevertheless, three-dimensionality in itself does not necessarily prove easier navigation, unless the visualisation and user interface suit the navigation tasks.

We have asserted goals and problems for navigation with mobile 3D maps, concentrating on manoeuvring in urban environments. The problems have been identified and a model has been presented as a solution framework. Interaction guidelines have been provided for 3D navigation. Using common navigation tasks as cases, we have applied these guidelines to yield a collection of interaction designs. 3D navigation is a complex problem and design solutions can be contradictory. Navigation efficiency is also highly context sensitive. An optimal 3D user interface is always a compromise, but we believe that the designs presented here lead to a positive user experience. Our future work concerns testing these solutions in the field.

It may seem that many of the challenges can be solved by technological advances. For example, urban positioning may be based on WLAN technologies or artificial GPS signal generators. 3D hardware will speed up rendering, and may release resources for better I/O management. However, GPS positioning may not be accurate or reliable in urban canyons, software-based rendering speed even with an optimised 3D engine may not suffice, and interface technologies such as mobile touch screens may not function perfectly. In any case, we are heading toward better solutions which will eventually enable creating host of new applications for urban pedestrians.

Acknowledgements

The m-LOMA 3D map project has been supported by EU Interreg IIIA. The Academy of Finland supported the work of the second author. We thank the lead programmers Ville Helin and Nikolaj Tatti. We also thank Sara Estlander who helped in the proofreading of this manuscript.

References

Bowman, D., Koller, D., and Hodges, L. (1997): Travel in immersive virtual environments: An evaluation of viewpoint motion control techniques. In *Proceedings of VRAIS'97*, pp. 45-52.

Buchholz, H., Bohnet, J., and Döllner, J. (2005): Smart and physically-based navigation in 3D geovirtual environments. In *Proceedings of Information Visualization 2005*, IEEE, pp. 629-635.

Burigat, S., and Chittaro, L. (2005): Location-aware visualization of VRML models in GPS-based mobile guides. In *Proceedings of the 10th International Conference on 3D Web Technology (Web3D 2005)*, New York: ACM Press, pp. 57-64.

Darken, R.P., and Sibert, J.L. (1996): Navigating large virtual spaces. *International Journal of Human-Computer Interaction*, Vol. 8, pp. 49-71.

Downs, R., and Stea, D. (1977): *Maps in Minds.* New York: Harper and Row.

Evans, G.W. (1980): Environmental cognition. *Psychological Bulletin,* Vol. 88, pp.259-287.

Garsoffky, B., Schwan, S., and Hesse, F.W. (2002): Viewpoint dependency in the recognition of dynamic scenes. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, Vol. 28 No. 6, pp. 1035-1050.

Golledge, R.G. (1999): Human wayfinding and cognitive maps. In *Wayfinding behavior. Cognitive mapping and other spatial processes,* R.G. Golledge, Ed. Baltimore: John Hopkins University Press.

Hart, P., Nilsson, N., and Raphael, B. (1968): A Formal Basis for the Heuristic Determination of Minimum Cost Paths. *IEEE Transactions on Systems Science and Cybernetics,* Vol. 4 No. 2, pp. 100-107.

Hanson, A.J., and Wernert, E.A. (1997): Constrained 3D navigation with 2D controllers. *IEEE Visualization*, pp. 175-182

Hughes, S., and Lewis, M. (2000): Attentive Camera Navigation in Virtual Environments. *IEEE International Conference on Systems, Man & Cybernetics*.

Jul, S., and Furnas, G.W. (1997): Navigation in Electronic Worlds: A CHI 97 Workshop. *SIGCHI Bulletin*, Vol. 29 No. 4, pp. 44-49.

Kirsh, D., and Maglio, P. (1994): On distinguishing epistemic from pragmatic action. *Cognitive Science*, Vol. 18, pp. 513-549.

Kiss, S., and Nijholt A. (2003): Viewpoint adaptation during navigation based on
stimuli from the virtual environment. In *Proceedings of Web3D 2003*, New York: ACM Press, pp. 19-26

Laakso, K. (2002): Evaluating the use of navigable three-dimensional maps in mobile devices. An unpublished Master's Thesis, *Helsinki University of Technology*. Helsinki: Department of Electrical and Communications Engineering.

Levine, M. (1982): You-are-here maps: Psychological considerations. *Environment and Behavior*, Vol. 14 No. 2, pp. 221-237.

Levine, M., Marchon, I., and Hanley, G. (1984). The Placement and Misplacement of You-Are-Here Maps. *Environment and Behaviour*. Vol. 16 No. 2, pp. 632-656.

Lynch, K. (1960): *The Image of the City.* Cambridge: M.I.T.Press.

Smith, S.P., and Marsh, T. (2004): Evaluating design guidelines for reducing user disorientation in a desktop virtual environment. *Virtual Reality,* Vol. 8 No. 1, pp. 55-62.

Meng, L., and Reichenbacher, T. (2005): Map-based mobile services. In *Map-based mobile services – Theories, Methods and Implementations,* Meng, L., Zipf, T and Reichenbacher, T. (eds): Springer, pp. 1-10.

Mou, W., and McNamara, T.P. (2002): Intrinsic frames of reference in spatial memory. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, Vol. 28, pp. 162–170.

Norman, D. (1988): *The Psychology of Everyday Things.* New York: Basic Books.

Nurminen, A. (2006): m-Loma—a mobile 3D city map. In *Proceedings of Web3D 2006*, New York: ACM Press, pp. 7-18.

Plesa, M.A., and Cartwright, W. (2007). Evaluating the Effectiveness of Non-Realistic 3D Maps for Navigation with Mobile Devices. In *Mobile Maps,* Meng, L. and Zipf, A. (eds.)

Presson, C.C., DeLange, N., and Hazelrigg, M.D. (1989): Orientation specificity in spatial memory: What makes a path different from a map of the path? *Journal of Experimental Psychology: Learning, Memory, and Cognition*, Vol. 15, pp. 887–897.

Oulasvirta, A., Nurminen, A., and Nivala, A-M. (submitted): Interacting with 3D and 2D mobile maps: An exploratory study.

Oulasvirta, A., Tamminen, S., Roto, V., and Kuorelahti, J. (2005): Interaction in 4-second bursts: The fragmented nature of attentional resources in mobile HCI. In *Proceedings of the 2005 SIGCHI Conference on Human Factors in Computing Systems (CHI 2005)*, New York: ACM Press, pp. 919-928.

Rakkolainen, I., and T. Vainio (2001): A 3D city info for mobile users. *Computers and Graphics, Special Issue on Multimedia Appliances*, Vol. 25 No. 4, pp. 619-625.

Roskos-Ewoldsen, B., McNamara, T.P., Shelton, A.L., and Carr, W. (1998): Mental representations of large and small spatial layouts are orientation dependent. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, Vol. 24, pp. 215–226.

Sorrows, M.E., and Hirtle, S.C., (1999): The nature of landmarks for real and electronic spaces. In *Spatial Information Theory*, Freksa, C. and D.M. Mark, Eds. Lecture Notes in Computer Science, Vol. 1661. Berlin: Springer, pp. 37-50.

Stuart, R. (1996): *The Design of Virtual Environments.* McGraw-Hill.

Vainio, T., and Kotala, O. (2002): Developing 3D information systems for mobile users: Some usability issues. In *Proceedings of the The Second Nordic Conference on Human-Computer Interaction (NordiCHI'02)*, New York: ACM Press.

Waller, D.A. (1999): An assessment of individual differences in spatial knowledge of real and virtual environments. *Unpublished doctoral dissertation*, Washington D.C.: University of Washington.

Wang, R.F., and Brockmole, J.R. (2003): Human navigation in nested environments. *Journal of Experimental Psychology: Learning, Memory, and Cognition,* Vol. 29 No. 3, pp. 398-404.

Witmer, B.G., Sadowski, W.J., and Finkelstein, N.M. (2002): VE-based training strategies for acquiring survey knowledge. *Presence: Teleoperators and Virtual Environments*, Vol. 11, pp. 1–18.

Woods, D.D., and Watts, J.C. (1997): How not to have to navigate through too many displays. In *Handbook of Human-Computer Interaction, 2nd edition*, Helander, M. G., Landauer, T. K. and Prabhu, P., Eds. Amsterdam: Elsevier Science.

Wraga, M. (2003): Thinking outside the body: An advantage for spatial updating during imagined versus physical self-rotation. *Journal of Experimental Psychology: Leaning, Memory, and Cognition*, Vol. 29 No. 5, pp. 993-1005.

11 PDA-Assisted Indoor-Navigation with Imprecise Positioning: Results of a Desktop Usability Study

Hartwig H. HOCHMAIR

Department of Geography, Saint Cloud State University

Abstract. Although most of today's navigation systems are used for guidance of cars, recent progress in mobile computing made it possible for research and industry to develop various prototypes of indoor-navigation systems in combination with PDAs. Independent of the presentation mode of route instructions, it is desirable that such real-time route guidance system automatically delivers the correct piece of information to the user at the right time. This requires that the PDA knows the user's position and orientation, which is not always available due to technical limitations of indoor sensing and positioning techniques, and potential signal dropouts. Using a desktop usability study, this chapter extends previous work on route instructions with mobile devices. The study explores the preferred modes of interaction between user and PDA in case of diluted position and orientation accuracies.

11.1 Introduction

While navigation systems for cars have already been commercialized for several years, the design of mobile navigation systems for indoor-navigation is still a relatively new research direction and one of the current challenges in the field of mobile mapping. A PDA-based application needs to provide accurate route instructions on small, low resolution interfaces in real time, which requires use of intelligent map generalization algorithms and choice of appropriate data sets for map visualization and smooth zooming functionality (Agrawala and Stolte, 2001; Sester and Brenner, 2004). Pedestrians have more route choices than for example car drivers, as their locomotion is not bound to lanes or affected by restrictions for car drivers (Corona and Winter, 2001). Due to the higher density of decision points for pedestrians, which is true especially for indoor-environments, a pedestrians guidance system requires precise information on the user's current position and orientation to provide the correct instruction on time. However, current positioning techniques do not always provide the required accuracy. A large body of research exists that focuses on determining the best mode of presenting route instructions to the users of mobile devices, whereas the interrelation between positioning accuracy and the best mode of route instructions is hardly reported in literature.

This chapter discusses the findings of an empirical study that examines how the interaction between user and PDA should be adapted if the PDA loses the signal required for precise indoor-positioning and/or orientation. In the discussion it is assumed that the optimal route has already been pre-computed and planned by the PDA.

Thus the focus is on route following, and the process of route selection is left aside. Section 11.2 reviews existing literature on modes of route instructions, and how route instructions are adapted to positioning accuracy. Section 11.3 presents the working hypotheses and the setup of the desktop study. This is followed by the presentation of the results and a discussion in section 11.4, and the conclusions and directions for future work in section 11.5.

11.2 Previous work

This section provides an overview of the various presentation modes for route instructions, positioning techniques, and resource-adaptive user-interface designs, as far as they are relevant for the explanation of the desktop usability study.

11.2.1 Presentation modes of route instructions on PDAs

Independent of the instruction mode, successful navigation requires the user to execute route instructions at the right time during the trip, i.e., neither too early nor too late. Desktop-based route planners are often utilized by printing a complete list of route instructions for a trip beforehand, so that the navigator has to synchronize with the printed route instructions during the trip. As opposed to this, mobile electronic route guidance systems with real time positioning capabilities communicate route instructions quasi simultaneously to the user and synchronize automatically with the user's current position. This method reduces the user's cognitive effort, but requires precise real-time positioning of the mobile device.

Route instructions can be provided through numerous presentation forms. Tversky and Lee (1999) found that both route maps and route descriptions can be decomposed by the same elements, and that verbal and graphic elements can be mapped onto one another. Experiments showed that navigators do not want to be given 'complete' route directions with reference to every single segment along the route, but rather a simplified set of textual route instructions (Lovelace et al., 1999; Klippel et al., 2003). Spatial chunking of elementary route segments into higher order elements can help to simplify route directions. Geldof and Dale (2002) suggest hierarchical structuring of route instructions through landmark- and path-based segmentation in order to hide redundant, i.e., irrelevant, route instructions from the user. Ideally, what remains are route directions that contain the pragmatic information content only (Frank, 2003). These design issues are relevant for route instructions in general and not restricted to PDA-assisted navigation. They are, however, taken into account for the formulation of the sample instructions used in the desktop study of this book chapter.

Literature provides numerous reports about empirical studies on route instructions for hand-held devices. In a field study on navigation guidance for pedestrians Reichl (2003) tested the usability of four modes of route instructions on an iPAC, namely written text, spoken text, map with start and destination only, and map with start, destination, and drawn route. The results showed that both text and map were considered as useful representation modes, where a small preference was given to maps. A similar result was found by Ceaparu et al. (2001), namely a slight, non-significant preference of maps over textual route instructions for indoor-navigation. The study of Reichl (2003) further revealed that spoken instructions were less preferred, as the candidates had to listen to the verbal instruction at least twice. Subjects were further asked to evaluate some fictive scenarios with other forms of route presentations. The combination between map with drawn route and voice was not considered as useful, as the map would provide all information required. Images were considered helpful for providing reorientation information at the beginning of the trip, as they would be more informative than written instructions for reorientation.

Rakkolainen et al. (2000) compared the usability of 2D maps and 3D VRML models in route guidance. The authors concluded that participants preferred the two representation modes together rather than any of them alone, and that the 3D model allowed users to recognize landmarks easier than this was the case with 2D maps. In a pilot study on mobile devices Kray et al. (2003) found that both route finding and initial orientation was slower with 3D models than with 2D maps. In the same study participants stated that 2D maps were sufficient for the given navigation task. Hurtig and Jokinen (2006) present a multimodal route navigation system for PDAs that combines speech, pen, and graphics. Wasinger et al. (2003) describe a navigation guidance system for pedestrians that combines 2D and 3D graphics with synthesized speech generation and fusion of speech and gesture input.

11.2.2 Indoor positioning methods

In general, location sensing techniques are imprecise, and precise location cannot be determined (Duckham et al., 2003). This is true both for outdoor and indoor navigation. Global Positioning (GPS) technology has become established as the most preferred method for outdoor positioning for mobile computers and mobile mapping, and it has been augmented with dead reckoning techniques (Ladetto et al., 2001; Randell et al., 2003) and network based location sensing (Djuknic and Richton, 2001). However, GPS and cellular-network-based positioning are not appropriate for indoor use due to loss of line-of-sight as well as signal blockage, fading and shadowing (Kolodziej, 2004).

Indoor positioning systems can be classified after the location of computation (Hightower and Borriello, 2001). The first group uses localized location computation (LCC) where the object being located computes its own location. For example, the MIT CRICKET system utilizes a series of ultrasound emitters to create the infrastructure. This system implements a local coordinate system using four active beacons instrumented with known positions in space. The beacons broadcast information on a radio frequency (RF) channel sensed by compass receivers. In this sense, the MIT CRICKET system behaves as a form of indoor GPS. The advantage of LLC techniques includes privacy and decentralized management, while one of the disadvantages is the computational burden placed on the mobile devices.

The second, more common group of systems, requires the located object to periodically broadcast, respond with, or otherwise emit signals to allow the external network infrastructure to locate it without directly involving the object in the computation.

For example, the Active Bat location system, developed by AT&T, requires users to carry Active Bat tags. In response to a request that the controller sends via short-range radio, a Bat emits an ultrasonic pulse to a grid of ceiling-mounted receivers. A central controller receives the distance measurements of the ceiling sensors and performs the lateration computation. Placing the burden on the infrastructure reduces the computational load of the mobile device. Other examples of systems that employ external infrastructures are SmartLOCUS (HP Labs), UbiTags (UbiSense), or RADAR (Microsoft research).

Hightower and Borriello (2001) and Kolodziej (2004) give an extensive review of location sensing technologies both for indoor and outdoor positioning. Table 11.1 lists some indoor positioning systems with their accuracy and precision. Accuracy refers to the grain size of the position information provided, whereas precision describes the percentage of how often such accuracy is provided. Scale denotes the coverage area per unit of infrastructure.

Table 11.1. Indoor positioning systems (1 ft = 30.48cm) (after Hightower and Borriello, 2001 and Kolodziej, 2004)

Positioning System	**Accuracy and Precision**	**Deployment and Scale**
CRICKET	4 x 4 feet regions (~100%)	about 1 beacon per 16 ft^2
RADAR	3-4.3 m (50%)	3 bases per floor
Smart LOCUS	< 20 cm (precision unknown)	Nodes placed every 2-15 m
Active Bats	9 cm (95%)	1 base per 10 m^2
UbiTags	30 cm (95%)	4 sensors for up to 400 m^2 1 UbiTag per object

Although positioning techniques are able to locate mobile devices at an accuracy of less than a meter, there is always a chance of signal dropout, or dilution due to poor lighting, close distance to ferrous materials, or noise in inertial systems, which causes position drift and uncertainty. Sensor fusion seeks to improve accuracy and precision by integrating many positioning systems to form hierarchical and overlapping levels of resolution. Statistically merging error distributions is an effective way to assess the combined effect of multiple sensors. Such methods are particularly important to bridge differences between different sensors in different environments, as different positioning techniques between outdoor and indoor environments should remain unnoticed by the user of a navigation system (Baus et al., 2005).

Baus et al. (2001) propose a taxonomy of resource adapted route presentation modes (Fig. 11.1). Which mode is used for route instructions depends on the precision of the PDA's knowledge about its location and orientation. If precise location and orientation are available a direction arrow can be produced (A). With a deteriorated orientation precision, a simple arrow would be misleading, and a graphical way description with a topological diagram should be used (B). If the position resolution is worse than room-size and covers several decision points, a clipped area of the surrounding including landmarks is displayed on the PDA display (C). To resolve ambiguity of location, the user can click his or her current position on the map. If there is only very rough or no information about the actual position and orientation, a greater portion of the map needs to be chosen, including global landmarks, such as stairs or elevators (D). Again the user can communicate his or her position to the system by clicking on several grey dots indicating the user's potential locations. In case of lost orientation, the system could regain the user's orientation by advising the user to reorient himself or herself towards a landmark. For outdoor navigation, the paper suggests that the ego-perspective mode makes only sense if the system has precise position and orientation information, and that in case of inferior quality, the system should switch to birds-eye perspective.

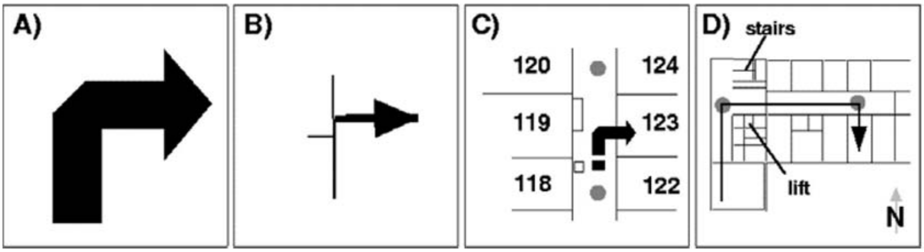


Fig. 11.1. Four different resource adapted graphical way descriptions in (Baus et al., 2001)

The study which is introduced in the next section uses simplified gradations of positioning and orientation errors. More specifically, a distinction will only be made between erroneous and error-free positioning and/or orientation.

11.3 Desktop usability study

The goal of this study was to examine how a user would prefer to interact with the guidance system in case the PDA is incapable of gaining precise positioning and/or orientation information during indoor-navigation. The study was set up on a desktop computer. The participants were shown 22 fictive indoor-navigation scenarios and had to rate several provided modes of interaction to handle the situation of imprecise positioning and orientation.

The choice of a simulated desktop study instead of a field study with a PDA device had several pragmatic reasons. First, no working PDA-positioning technology is actually available at the department where the study was run. A possible workaround could be a setup where the experimenter marks the simulated detected user position and orientation on another device, which requires only working network connections. Even then it would be difficult to examine the user's most preferred mode of interaction as the different options could only be shown one after the other on the small PDA display, which would make it difficult for the participant to memorize and evaluate the various options. Second, to give a realistic situation with a PDA, the participant would have to walk the pre-defined path for each scenario, i.e., a total of 22 times, and then evaluate the options for each scenario. One can expect that participants would lose concentration during traversing the same path 22 times and flipping through all suggested options after each traverse. Decreasing the number of scenarios per participant, however, would not allow us to test paired preferential scores of individuals for assess the impact of the independent variables of interest (user position, signal error type, default display mode).

It is difficult to tell whether a field study and a desktop study yield similar results. To the author's knowledge there exists no literature that analyzes potential differences between results of a simulated desktop survey and a field study with PDAs. Numerous studies have, however, compared wayfinding behaviour between real and virtual environments. No clear tendency can be found in the comparisons. Whereas, for example, Haq et al. (2005) report on observed differences between wayfinding performance in a real hospital and its virtual counterpart, and Weiss and Jessel (1998) point out the differences in the wayfinder's experience when learning to navigate in a virtual environment as opposed to the real world, a recent study found that the performance of a complex navigational task in a virtual environment was comparable to the real world (Ruddle and Lessels, 2006).

11.3.1 Participants

Most of the 23 participants (9 female, 14 male) were students of the geography program at Saint Cloud State University. Participation was volunteer, and participants received a small payment for their participation.

11.3.2 Hypotheses

Hypothesis 1. Users want to interact in case of signal loss before turns only.

The first prediction is based on the fact that navigators prefer not be overloaded with instructions that are not required for proceeding in the correct direction. Route instructions are typically needed along unknown portions of the route, before a turn at an unknown decision point, or to get confirmation for the correctness of the current walking direction (Michon and Denis, 2001; Geldof and Dale, 2002; Klippel et al., 2003). The assumption related to this hypothesis is that the user of a PDA does not want to take additional steps of interaction with the PDA during navigation, even if there is a signal dropout, as long as the user is on track and no turn needs to be made. Thus it is expected that, in the case of a signal dropout, participants will be more open to interact with the system if they are close to a decision point where they are supposed to turn than when they are somewhere within a long straight segment of the route.

Hypothesis 2. Users stick to the current representation mode in case of signal loss.

Any change of the instruction mode during navigation imposes additional cognitive effort on the user. Thus it can be expected that, in case of diluted positioning accuracy, users of a PDA prefer to stick to the current instruction mode. If, for example, a 2D map has been used to display route instruction before the PDA loses positioning, the user would prefer an interactive 2D map to resolve that problem. This hypothesis will be tested for instruction modes that provide enough information about the surrounding environment so that the user could manually synchronize his or her current position with the instruction on the PDA. These information-rich modes do not necessarily require information about the user's orientation to provide useful route instructions.

Therefore, the scenarios related to this hypothesis are tested for loss of positioning accuracy only.

Hypothesis 3. The type of signal error impacts the preferred mode.

Using the *Direction Arrow* mode (Baus et al., 2001) or the *2D Route Sketch* mode (Kray et al., 2003) (see Table 11.2a, b) require the PDA to know the navigator's position and heading precisely. Although these two modes afford permanent attention, because decision points are not chunked in the instructions, these modes can be important in situations of distress, such as a fire emergency, where each instruction should be kept as simple as possible. This can be achieved by not forcing the user to count decision points (e.g., turn right into the fourth door) or to remember landmarks that are not visible from the current position (e.g., walk until you see the elevator to your left). However, in case of signal loss, it is not possible for the user to manually synchronize her current position with the route instruction when scrolling through the instructions, because the two suggested modes provide only limited information about the surrounding environment. Thus the PDA needs to switch to a more information-rich mode in the case of a signal dropout. The third hypothesis is that the user's preferred mode of interaction, in case of a signal dropout while using the *Direction Arrow* or *2D Route Sketch* mode, depends on the type of positioning error (position, orientation, or both).

11.3.3 Setup of the study

The desktop study was realized as a sequence of Web pages opened on an Internet browser. A server-sided log file recorded the participants' preference statements.

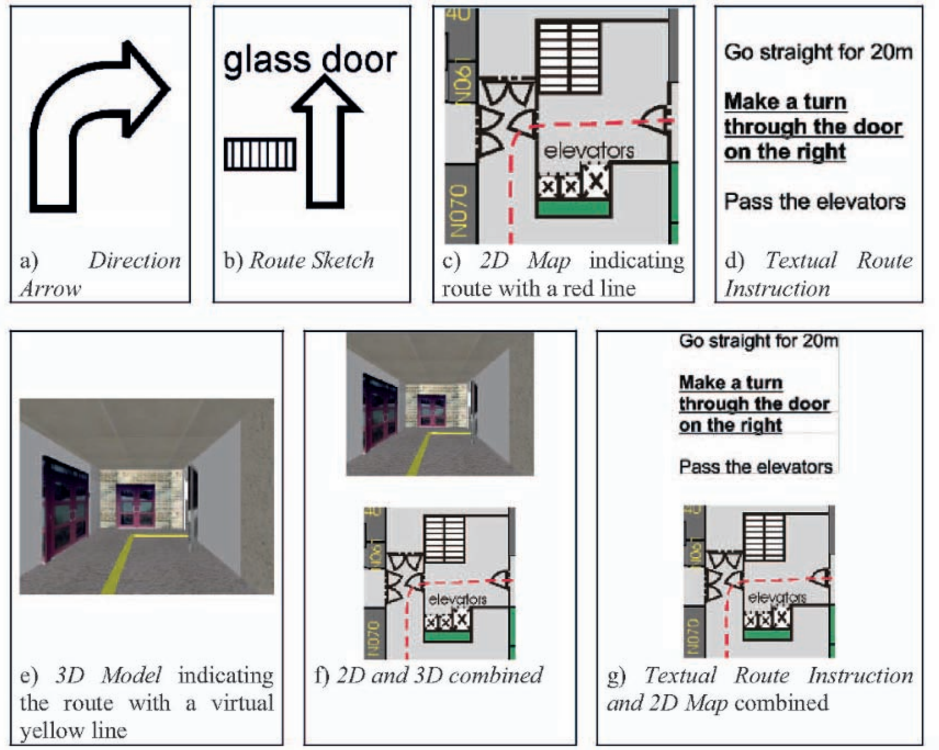
At the beginning of the study, an introductory page explained the seven default modes of PDA route instructions that were simulated in the study (Table 11.2). It included those modes that were frequently discussed in literature (see section 11.2.1). The seven modes include guidance by direction arrow (a), route sketch (b), 2D map (c), textual route instructions (d), 3D model (e), a combination of 2D map and 3D model (f), and a combination of textual instructions and 2D map (g).

The simulated scenarios visualize a part of the ground floor of the "Naturwissenschaften 1"-building at the University of Bremen. This building was chosen as the author is familiar with its environment, and as the floor maps of the university buildings are freely available (http://www.uni-bremen.de/lageplan). The 2D maps used in the study were adapted from the original floor maps. The 3D scenes were designed with GtKRadiant 1.5.0, courtesy of Id Software Inc. The text mode showed instructions that were spatially chunked and which were therefore referring to turning points or confirmation points only.

The second introductory page explained the three error types that would be simulated in the 22 scenes, namely positioning error, orientation error, and a combination of the two (Table 11.3).

In the study, the participants would be shown a 2D overview map of the situation on the left side on the screen, together with a fictive user's PDA display for that situation on the right. The instructions explained the symbols used to visualize the various types of errors in the 2D overview map, and the corresponding error messages on the PDA display.

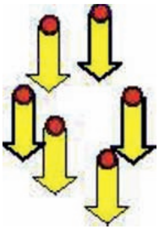
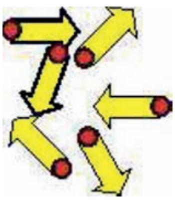
Table 11.2. Seven modes of route instruction for the PDA used in the study



a) *Direction Arrow*; b) *Route Sketch*; c) *2D Map* indicating route with a red line; d) *Textual Route Instruction*; e) *3D Model* indicating the route with a virtual yellow line; f) *2D and 3D combined*; g) *Textual Route Instruction and 2D Map* combined

The third introductory page used a scenario to explain the combination of all components as they were used for each scenario of the study (Fig. 11.2). Each scenario includes a 2D overview map of the building with the PDA's believed (i.e., potentially erroneous) position and heading of the fictive navigator. A dashed line depicts the path travelled so far by the fictive navigator and the remaining path planned by the PDA. To the right, a PDA display visualizes the default instruction mode for that scenario. Next to this, another display shows the error message that pops up and corresponds to the error type on the 2D map. Thus the participant would be given the 2D overview map, the default display mode, and an error message on the PDA, where the 2D map was only visible to the participant for the purpose of the study. A "real" user of the route guidance system would not have access to such an overview map. During the study the participant would be asked to rate a list of given interaction options regarding the usefulness of each option for a fictive user that receives an error message of the signal dropout but has no 2D overview map.

Table 11.3. Visualization of orientation error, position error, and combined error in a 2D overview map and the PDA display

2D overview Map: error symbols	2D overview Map: explanation in introductory text	PDA: error message
	The PDA knows the user's **position**, but does not know the **direction** in which the user is heading.	
	The PDA lost the user's **position**, but knows the **direction** in which the user is heading. The range of guessed positions is indicated by red dots.	
	The PDA lost the user's **position** and **orientation**. The range of guessed positions is indicated by red dots.	

In the sample scene that was presented on the introductory page (Fig. 11.2), the PDA uses the *Direction Arrow* default mode. The system detects an orientation error, which is indicated through the wind rose symbol in the 2D map and the corresponding error message in the right PDA display.

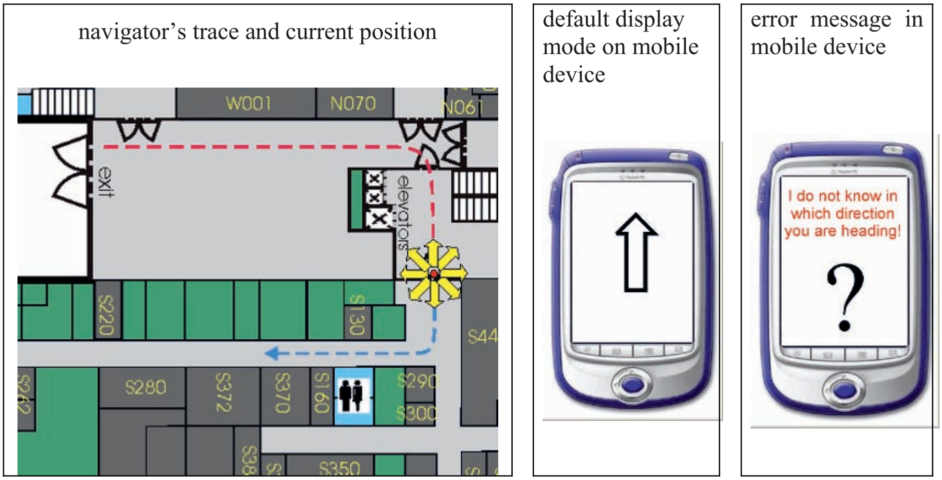


Fig. 11.2. Preview of one scenario on the introductory page

11.3.4 Selection of scenes

In order to test the three hypotheses, three independent variables are varied between the scenarios, i.e., navigator's position where a signal loss is encountered, default display mode, and type of error. To test the impact of the navigator's position on the willingness to interact with the PDA (hypothesis 1), the position of signal loss is varied between before a right turn (as visualized in Fig. 11.2) and the middle of a long straight segment (a spot near the head of the dashed arrow in Fig. 11.2). The first of the two locations is referred to as "turn", and the second one as "straight".

Besides these two options, the scenes varied also between the seven default display modes introduced in Table 11.2, and the three error types in Table 11.3. The total number of possible combinations amounts therefore to 2*7*3=42, which is a too high number of scenes for a single participant to evaluate. In order to reduce the number of scenes, following considerations were made: As the *Direct Arrow* mode and the *Route Sketch* mode require knowledge both about the navigator's position and orientation, all 12 scenes related to any of these two display modes were kept to test the impact of all possible types of errors. All other five display modes (*2D Map, 3D Model, Textual Route Instruction, 2D and 3D combined, Textual Route Instruction and 2D Map combined*) can provide useful route guidance with position information only. For example, even with an orientation error in the 2D map mode, the system can automatically scroll the clipped map and centre it around the user position. The system is also able to rotate and align the centred map along with the navigator's current walking direction. A lack of orientation information would only then require user interaction, if the system was not capable of catching the user's trajectory. Therefore, orientation error and combined error are not specifically discussed for these five modes. Together with two possible locations (*turn* and *straight*), these considerations yield a total of 22 scenes for each participant (Table 11.4). For each scene a set of interaction modes to handle signal loss was suggested and presented to the participants for evaluation.

Table 11.4. Parameter values for the selected 22 scenes (*pos* = positioning error, *or* = orientation error, *comb* = combined error)

	Turn			Straight		
	pos	**or**	**comb**	**pos**	**or**	**comb**
Direction Arrow	x	x	x	x	x	x
Route Sketch	x	x	x	x	x	x
2D Map	x			x		
3D Model	x			x		
Textual Route Instruction	x			x		
2D Map and 3D Model	x			x		
Textual Route Instruction and 2D Map	x			x		

11.3.5 Options for interaction in the case of a signal loss

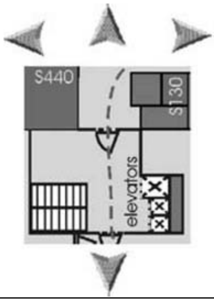
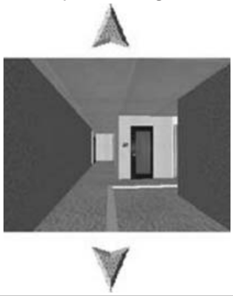
Based on some pre-testing, and a review of related literature, a set of 10 options for handling signal dropout was selected to be shown for evaluation together with a given scene. The options included manual scrolling of a 2D map, a 3D visualization, a sequence of text instructions, pinpointing current position and/or heading on a 2D map, telling the PDA one's current position and/or heading in natural language, using the PDA camera to detect current position and/or orientation, and following the PDA's verbal instructions to re-orientate oneself in case of lost orientation (Table 11.5).

The 10th option (not shown in Table 11.5) was a textual statement suggesting that the PDA would not request any interaction from the user: "*The PDA does not do anything else besides informing the user about the error, showing the last instruction screen, and keeping on trying to receive a location/orientation signal*". This option, which is referred to as "do nothing", was provided for all 22 scenes.

Not all options are meaningful for all scenes. Options 5, 6, and 9 cannot be applied for scenes with a positioning error only. Options 4 and 6 are inappropriate for scenes with an orientation error only, and options 4, 5, and 9 provide insufficient information if the system lacks knowledge about both the navigator's position and heading. Based on these considerations, each scene was provided with only between 7 and 8 options for evaluation.

Each of the 22 screens in the study showed the scenario on top of the page (including the 2D map, the default display mode, and the error message on the mobile device), and the appropriate options in a tabular-like form (similar to Table 11.5) with a pulldown menu next to each option for evaluation (see Fig. 11.3). The participants could assign each option between 1 and 10 points, where 1 point was characterized as "useless or complicated suggestion", and 10 as "very useful and convenient suggestion". In addition to that, participants could write some comments in a text field for each scene. The 22 screens were shown in a different random order to each participant. No time limit was given for completing the evaluations. The average time for evaluation of all 22 scenes was 26.0 minutes (stddev=10.0), excluding the time for reading the three introductory pages.

Table 11.5. 9 out of 10 suggested options of interaction in the case of a signal dropout. Figures and explanations are as shown to the participant.

1) Switch to 2D map which the user can manually scroll until the PDA catches position again. The map aligns with the path. 	2) Switch to 3D mode which allows the user to scroll manually back and forth along the virtual yellow path-line in 3D until the PDA catches position again. 	3) Switch to text-mode, which allows the user to scroll manually through instructions, until the PDA catches position again. 
4) Indicate on a 2D map on the PDA where you are, and wait for further instructions from the PDA. 	5) Indicate on a 2D map of the PDA which direction you are looking at, and wait for further instructions from the PDA. 	6) Indicate on a 2D map of the PDA your position and heading, and wait for further instructions from the PDA. 
7) Tell the PDA by voice in natural language your position and/or heading, and wait for further instructions from the PDA. 	8) Use the internal camera of the PDA to let it detect your position and/or heading, and wait for further instructions from the PDA. 	9) The system advises the user to reorient towards a nearby feature to regain orientation. 

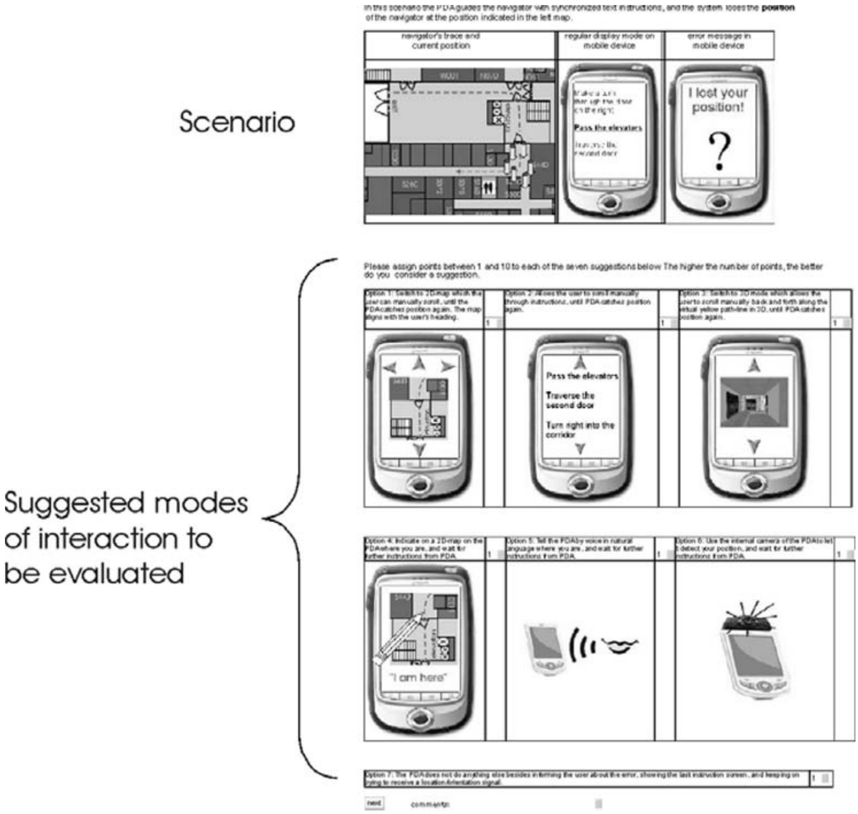


Fig. 11.3. Screenshot of the study setup: 2D map, default display mode, and simulated error message (upper block), and seven suggested modes to handle the positioning problem (lower block)

11.4 Results and discussion

In the analysis of the results it was tested whether the preferential behaviour formulated in the hypotheses in section 11.3.2 could be observed. Based on the independent variable that needed to be tested for its impact on the user preference (location, default presentation mode, type of error) participants' responses were grouped accordingly. The analysis compares either the preference for the same option between different scenes (for hypothesis 1), or it compares the preference between different options for one scene (hypotheses 2 and 3). The first method gives insight into the impact of location on the preferred interaction mode in the case of signal loss, whereas the latter method is used to assess the impact of default mode and error type on the user's preferred interaction mode.

11.4.1 Hypothesis 1: impact of user location

To test whether the user is less willing to interact with the system on straight segments than before turning points, all scenarios were grouped regarding the user's fictive location at signal loss. For each group, the number of points that were on average assigned to option 10 ("do nothing") were extracted. Although that number was, as expected, higher for the "straight" location (mean = 2.48) than for the "turn" location (mean = 2.26), the difference between the two locations was not significant. The low average number of points in both groups indicates that the meaning of this option was not clear to all participants. This option was only presented in textual form, as opposed to the other options which all included a pictorial representation of the display. This might have decreased the participants' attention for that particular option.

A second way to test hypothesis 1 was to make pairwise comparison between scenes that differ only in the simulated navigator's position, keeping all other parameters (display mode, error type) constant. This gave 11 pairs of scenes. For each of the two scenes in a related pair, the most preferred option was determined, which was followed by a comparison of the number of points assigned to that most preferred option between both scenes. Referring to Table 11.4, corresponding scenes between the left half ("turn") and the right half ("straight") were compared. A smaller response for the "straight" option than for the "turn" option could be interpreted as the respondent's smaller interest in solving the problem of signal dropout interactively for situations where it is not absolutely necessary. Results showed that one pair of scenes showed a significantly smaller preference value for the "straight" location than for the "turn" location ($mean_{straight} = 6.57$; $mean_{turn} = 7.48$; $p < 0.05$). This scene used the 2D map as default display, and the most preferred option to handle signal dropout was the 2D scrollmap (Table 11.5-1) in both related scenes. The results indicate that the position of signal loss affects, at least when using the 2D floor map, the user's willingness to interact with the system in the case of signal loss. Differences for other default display modes, such as the 3D model, were not significant.

11.4.2 Hypothesis 2: impact of default mode

To test this hypothesis, the most preferred option of error handling was identified for each of the 22 scenarios for each participant. The analysis should reveal whether or whether not the preferred option was similar to the default mode shown in that scene. For example, if the default mode was the 3D model, and the most preferred option in the case of signal loss was the scroll mode of the 3D model, these two modes were considerably similar and thus supporting hypothesis 2. If the most preferred option for that same situation was any other mode, such as the text mode, default mode and option were different, which would not support the hypothesis. In a next step it was tested whether the preferred option was assigned a significantly higher number of points than all remaining options of that scene. If so, this revealed a single favourite option to handle imprecise positioning for a given default mode. For all options the assigned points were found to be normal distributed among participants, except for option 10 ("do nothing"). Therefore a Paired Samples T-Test was used for comparing the mean number of points between any two options, except for option 10, where the Wilcoxon Signed-Rank test was used instead. Scenes using the first two modes, i.e., *Direction Arrow* and *Routesketch*, were excluded from the analysis, as these modes cannot be utilized in the case of signal loss. Thus 10 scenes were kept for analysis, namely the ones in the lower five rows of Table 11.4. As the scenes simulate a positioning error, all options referring to orientation errors (5, 6, and 9 in Table 11.5) were excluded for these scenes, leaving 7 options for the analysis.

The ten rows in Table 11.6 denote the ten scenes analyzed. The left most column shows the default display mode for each scene, the second-left column the simulated position of signal loss, and the third column the option that was on average the most preferred one to handle signal loss. The right-most seven columns list the significance levels (2-tailed) for differences between the most preferred option and the option listed in the column header. Numbers in boldface highlight a significance level > 0.05, which indicates these alternative options that are not significantly less preferred than the most preferred option.

Table 11.6. Preferred option (Pref) for given default mode (Default) and position (Pos) in case of signal loss.

	Default	***Pos***	**Pref**	**2D**	**3D**	**text**	**pencil**	**voice**	**cam**	**nothing**
1	***2D***	*turn*	**2D**	-	.002**	.003**	**.287**	.001**	.004**	.000**
2		*straight*	**pencil**	**.401**	.041*	.008**	-	.000**	.015*	.000**
3	***3D***	*turn*	**pencil**	**.352**	**1.000**	.030*	-	.002**	.018*	.003**
4		*straight*	**pencil**	**.111**	**.433**	.002**	-	.000**	.006**	.000**
5	***Text***	*turn*	**2D**	-	**.087**	**.476**	**1.000**	.001**	**.089**	.000**
6		*straight*	**2D**	-	.008**	.045*	**.893**	.002**	.008**	.000**
7	***2D+3D***	*turn*	**3D**	**.803**	-	.038*	**.805**	.001**	.032*	.000**
8		*straight*	**2D**	-	**.144**	.000**	**.353**	.000**	.005**	.000**
9	***Text+2D***	*turn*	**2D**	-	.001**	.008**	**.790**	.000**	.002**	.000**
10		*straight*	**2D**	-	.008**	**.071**	**.850**	.001**	.010*	.000**

** difference between means is significant at the 0.01 level (2-tailed)
* difference between means is significant at the 0.05 level (2-tailed)

In 5 cases (rows 1, 7, 8, 9, and 10) the most preferred option is similar to the default mode, indicating that users wants to keep the default display mode to handle signal loss. These cases support the proposed hypothesis. Further, in rows 2, 3, 4, and 5, no significant difference in preference has been identified between the most preferred mode and the mode most similar to the default mode, which also supports the hypothesis. The only exception is in row 6, where the preference between default mode (textual route instruction) and most preferred option (2D map) is significant. In this case, users seem to prefer to change from text mode to 2D map mode to resolve signal dropout.

The results suggest that voice recognition and use of camera for position detection are inappropriate modes to handle positioning uncertainty, and the number of points assigned to the two options was smaller than expected. A potential explanation for the refusal of voice or camera mode is that the participants could not imagine how these modes would work. Results might be different if these modes were tested with real PDAs. In the presented desktop study the acceptance of voice mode might also have been higher if route instructions by voice had been offered as one of the default modes in the PDA design.

Overall, the results in Table 11.6 reveal that generally scrollable 2D maps as well as a device for pinpointing one's location on a map (option *pencil*) are considered as convenient options for any default display mode in case of imprecise positioning.

Some of the participants' qualitative statements given during the study provide a better insight into these results. The comments suggest that more than one mode for re-positioning at a time might be useful and that the pinpointing option is useful only if the signal is caught again in time.

- "Some of the options take longer so I gave them a lower score because when you are lost, time can not be wasted."
- "A manual mode of some sort to re-establish position or direction is important, but there should be several options"
- "Option 4 (pencil) is all right, if I know where I am at, but if I can't get signal, it doesn't do any good."
- "I tend to stay away from anything that is voice activated because it often doesn't understand you anyway."
- "2D maps are great as long as they are orientated the same way you are facing."

11.4.3 Hypothesis 3: impact of error type

Display modes *c*-*g* in Table 11.2 do not depend on orientation information to provide useful route instructions. Further, these modes, if made interactive, give enough information to the user to re-position himself or herself through scrolling either in the 2D map, the 3D model, or the text instructions in the case of a diluted positioning signal. Preferences for these options have been discussed in section 11.4.2. Contrary to these modes, modes *a* and *b* in Table 11.2 (*Direction Arrow* and *Routesketch*) require both orientation and position information to provide useful instructions to the user. They provide hardly any information about the surrounding environment for the user to re-orient and re-position himself or herself, even in a scroll-mode. Therefore, the display mode needs to be changed in case of signal loss. The question related to the hypothesis is whether the type of error impacts the preferred option to handle signal loss for these two modes. 12 scenes were analyzed, namely the scenes checked in the first two rows of Table 11.4. It was tested whether the same options were preferred for scenes that share the same error type (positioning error, orientation error, combined error). Table 11.7 lists the most preferred options for the 12 scenes, and the level of significance (2-tailed) for differences between the most preferred option and the remaining options (Paired Samples T-Test).

The *pencil* option means that the user can pinpoint on a 2D map his or her location, or orientation, or both, depending on the type of error. Similarly, the *advise user* option (advise) depends on the type of error and means that the PDA gives the user written instructions to either re-orient, or to re-orient and re-position himself or herself. This option was not provided for the positioning error, assuming that this option was too complicated for re-positioning and to be inferior to the pinpointing method. For all scenes the "do nothing" option was significantly less desired than the most preferred function (p=.000) and is therefore excluded from the table.

Table 11.7. Preferred option (Pref) for *Direction Arrow* and *Route Sketch* default modes (Def) in case of signal loss for given error types (Error) and positions (Pos).

	Def	Error	Pos	Pref	2D	3D	text	pencil	voice	cam	advise
1	***Direction***	*Pos*	*turn*	**pencil**	**.506**	.086	.031*	-	.000**	.020*	N/A
2	***Arrow***		*str*	**2D**	**-**	.003**	.006**	**.672**	.001**	.010*	N/A
3		*Or*	*turn*	**advise**	**.174**	.009**	.007**	**.176**	.001**	.000**	-
4			*str*	**advise**	**.564**	.014*	**.059**	.012*	.000**	.003**	-
5		*Com*	*turn*	**pencil**	**.723**	.046*	.019*	-	.004**	.019*	**.261**
6			*str*	**advise**	**.547**	.048*	.023*	.433	.002**	.049*	-
7	***Route***	*Pos*	*turn*	**2D**	-	**.067**	**.136**	**.833**	.009**	.029*	N/A
8	***Sketch***		*str*	**3D**	**.281**	-	**.377**	**.786**	.001**	.023*	N/A
9		*Or*	*turn*	**advise**	**.062**	.009**	.000**	**.088**	.000**	.000**	-
10			*str*	**advise**	**.055**	.016*	.002**	.041*	.000**	.001**	-
11		*Com*	*turn*	**pencil**	**.286**	.002**	.007**	-	.000**	.001**	**.050**
12			*str*	**pencil**	**.511**	**.147**	.013*	-	.000**	.006**	.024*

** difference between means is significant at the 0.01 level (2-tailed)
* difference between means is significant at the 0.05 level (2-tailed)

The results suggest that each error type yields different preferred options to handle signal loss. In the case of a positioning error, a 2D scroll map is the most preferred option, followed by pinpointing the current location on the 2D map, and using a scrollable 3D model. If the user's orientation is unknown, the most preferred option is to be advised by the PDA to orient towards a known object.

If both position and orientation are unknown, the participants feel most comfortable with pinpointing location and orientation on a 2D map. Only minor differences were observed across the two locations for each scene, and the two default modes.

Although the pinpointing method is often preferred in case of a position error, it is questionable how practical such a solution is, as the PDA might not catch a positioning signal right away, and therefore not be able to update instructions on the following route segments. In this respect, a scrollable 2D map or 3D model, or scrollable textual instructions seem to be more usable until the necessary signals are received again.

11.5 Conclusions

A desktop-based usability study was conducted to assess the preferred interaction modes of PDA users in case of a signal dropout during indoor-navigation. More specifically, the role of the navigator's position regarding turning points, the default display mode of route instructions, and the error type were examined. In concordance with hypothesis 1 it was found that the user is less willing to interact with the PDA when being located further away from the next turn. This finding was significant only for the 2D map mode. With one exception, the expectations of hypothesis 2 were met, namely that a navigator prefers not to change the instruction mode in the case of signal loss. For the two instruction modes that include only marginal information about the environment, namely *Direction Arrow* and *Route Sketch*, the preferred mode of interaction was found to depended on the type of error, thus supporting hypothesis 3.

The study was performed on a desktop environment, which is sufficient to get a coarse assessment of user preferences in case of signal loss. For more detailed feedback on user interface design in such situation, it will be necessary to evaluate the proposed options in a real indoor-environment together with PDA hardware. As both reliable indoor positioning techniques and proposed interaction methods (e.g., voice recognition, camera detection, or automatic map orientation) are still under development, a corresponding field study would be technically challenging. This is especially true when generating controlled signal-dropouts or dilution of signals, where the mobile system should react as predicted, and the optimal map scale for displaying the scroll-map in the case of a signal loss would have to be found. Using a field study would also reveal at which point during imprecise positioning a user might turn away from technology and ask a human for directions.

Acknowledgements

The author thanks the Transregional Collaborative Spatial Cognition Research Center (SFB/TR8) for providing the server infrastructure used in the study. Jennifer Fu provided valuable feedback on the design of the survey.

References

Agrawala, M., and Stolte, C. (2001): Rendering effective route maps: improving usability through generalization. In *Proceedings of Siggraph 2001*, E. Fiume, Ed. New York, NY, USA: ACM Press, pp. 241 - 249.

Baus, J., Kray, C., Krüger, A., and Wahlster, W. (2001): A Resource-Adaptive Mobile Navigation System. In *Proceedings of the International Workshop on Information Presentation and Natural Multimodal Dialogue (IPNMD-2001).* Verona, Italy: ITC-IRST, pp. 5-9.

Baus, J., Krüger, A., and Stahl, C. (2005): Resource-Adaptive Personal Navigation. In *Multimodal Intelligent Information Presentation*, O. Stock and M. Zancanaro, Eds. Berlin: Springer, pp. 71-93.

Ceaparu, I., Thakkar, P., and Yilmaz, C. (2001): *Visualizing Directions and Schedules on Handheld Devices*. Maryland, Department of Computer Science, University of Maryland College Park. http://www.cs.umd.edu/class/fall2001/cmsc838s/RoverAVW.doc

Corona, B., and Winter, S. (2001): Datasets for Pedestrian Navigation Services. In *Proceedings of the AGIT Symposium*, J. Strobl, T. Blaschke and G. Griesebner, Eds. Salzburg, pp. 84-89.

Djuknic, G. M., and Richton, R. E. (2001): Geolocation and Assisted-GPS. *IEEE Computer*, Vol. 34 No. 2, pp. 123-125.

Duckham, M., Kulik, L., and Worboys, M. F. (2003): Imprecise navigation. *GeoInformatica*, Vol. 7 No. 3, pp. 79 - 94.

Frank, A. U. (2003): Pragmatic Information Content: How to Measure the Information in a Route Description. In *Perspectives on Geographic Information Science*, M. Duckham, M. Goodchild and M. Worboys, Eds. London: Taylor and Francis, pp. 47-68.

Geldof, S., and Dale, R. (2002): *Improving route descriptions on mobile devices*. ISCA Workshop on Multi-modal Dialogue in Mobile Environments, Kloster Irsee, Germany.

Haq, S., Hill, G., and Pramanik, A. (2005): Comparison of configurational, wayfinding and cognitive correlates in real and virtual settings. In *5th International Space Syntax Symposium, Vol. II*, pp. 387 - 405.

Hightower, J., and Borriello, G. (2001):Location Systems for Ubiquitous Computing. *IEEE Computer*, Vol. 34 No. 8, pp. 57-66.

Hurtig, T., and Jokinen, K. (2006): *Modality Fusion in a Route Navigation System*. Workshop on Effective Multimodal Dialogue Interfaces, Sydney, Australia.

Klippel, A., Tappe, H., and Habel, C. (2003): Pictorial representations of routes: Chunking route segments during comprehension. In *Spatial Cognition III*, C. Freksa, W. Brauer, C. Habel and K. Wender, Eds. Berlin: Springer, LNAI 2685, pp. 11-33.

Kolodziej, K. (2004). Indoor location technology opens new worlds. *GEO World*, Vol. 6 No. 1.

Kray, C., Elting, C., Laakso, K., and Coors, V. (2003): Presenting Route Instructions on Mobile Devices. In *Proceedings of IUI'03.* New York, NY: ACM Press, pp. 117-124.

Ladetto, Q., Gabaglio, V., and Merminod, B. (2001): *Two Different Approaches for Augmented GPS Pedestrian Navigation*. International Symposium on Location Based Services for Cellular Users (Locellus), Munich, Germany.

Lovelace, K. L., Hegarty, M., and Montello, D. R. (1999): Elements of Good Route Directions in Familiar and Unfamiliar Environments. In *Conference on Spatial Information Theory (COSIT'99)*, C. Freksa and D. M. Mark, Eds. Berlin: Springer, LNCS 1661, pp. 65-82.

Michon, P., and Denis, M. (2001): When and Why Referring to Visual Landmarks in Direction Giving? In *Conference on Spatial Information Theory (COSIT'01)*, D. Montello, Ed. Berlin: Springer, LNCS 2205, pp. 292-305.

Rakkolainen, I., Timmerheid, J., and Vainio, T. (2000): *A 3D City Info for Mobile Users*. 3rd International Workshop on Intelligent Interactive Assistance and Mobile Multimedia Computing (IMC'2000), Rostock, Germany.

Randell, C., Djiallis, C., and Muller, H. L. (2003): Personal Position Measurement Using Dead Reckoning. In *Proceedings of Seventh International Symposium on Wearable Computers*, C. Narayanaswami, Ed. IEEE Computer Society, pp. 166-173.

Reichl, B. (2003): Potential verschiedener Präsentationsformen für die Vermittlung von Routeninformationen in Fußgängernavigationssystemen (FNS) (Diplomarbeit, Institut für Kartographie und Geo-Medientechnik, Technische Universität, Wien).

Ruddle, R. A., and Lessels, S. (2006): For efficient navigational search humans require full physical movement but not a rich visual scene. *Psychological Science*, Vol. 17, pp. 460-465.

Sester, M., and Brenner, C. (2004): Continuous Generalization for Visualization on Small Mobile Devices. In *Developments in Spatial Data Handling*, P. Fisher, Ed. Berlin: Springer, pp. 469-480.

Tversky, B., and Lee, P. U. (1999): Pictorial and Verbal Tools for Conveying Routes. In *Conference on Spatial Information Theory (COSIT'99)*, C. Freksa and D. M. Mark, Eds. Berlin: Springer, LNCS 1661, pp. 51-64.

Wasinger, R., Stahl, C., and Krüger, A. (2003): M3I in a Pedestrian Navigation & Exploration System. In *4th International Symposium on Human Computer Interaction with Mobile Devices (Mobile HCI 2003)*, L. Chittaro, Ed. Berlin: Springer, LNCS 2795, pp. 481-485.

Weiss, P., and Jessel, A. (1998): Virtual reality applications to work. *Work*, Vol. 11 No. 3, pp. 277-293.

12 Accuracy and Performance Assessment of a Window-Based Heuristic Algorithm for Real-Time Routing in Map-Based Mobile Applications

Hassan A. KARIMI[1], Peter SUTOVSKY[1], Matej DURCIK[2]

[1]School of Information Sciences, University of Pittsburgh
[2]SAHRA-HWR, University of Arizona

Abstract. The demand for routing algorithms that produce optimal solutions in real time is continually growing. Real-time routing algorithms are needed in many existing and emerging applications and services. An example is map-based mobile applications where real-time routing is required. Conventional optimal routing algorithms often do not provide acceptable real-time responses when applied to large real road network data. As a result, in certain real-time applications, especially those with limited computing resources (e.g., mobile devices), heuristic algorithms that can provide good solutions, though not necessarily optimal, in real time are employed. In this chapter, we present two approaches for limiting the search space using a window-based heuristic algorithm to compute shortest routes and analyze their solutions and performances using real road network data. The results of a set of experiments on the two approaches show that the window-based heuristic algorithm produces aceptable response times using real road network data and that window sizes and orientations impact accuracy and performance of the algorithm.

12.1 Introduction

Routing is a fundamental function in numerous map-based mobile applications. Example applications are navigation, location-based services, and automatic vehicle location. In such real-time map-based mobile applications, the overall accuracy and performance of the underlying routing algorithm is of particular interest. This is because routing accuracy influences the user's confidence on and routing time performance impacts the real-time response time of the maps produced. Providing a reasonable level of confidence on the routes (which in some applications are the only maps produced) to the users and having a fast routing solution (which improves response time to real-time activities), will enhance the usability of real-time map-based mobile applications. Therefore, it is imperative to realize the accuracy and time performance of routing algorithms and their impact on such applications.

Routing belongs to a class of problems widely known as optimization and is computed by using either conventional algorithms or heuristic algorithms. Conventional routing algorithms are suitable in applications where the network size is not large and there is no real-time processing constraint. However, in applications where the network size is very large and routes must be computed in real time, heuristic routing algorithms are considered, which may not guarantee optimal solutions. This is because conventional routing algorithms' performance lowers as the size of the network increases resulting in unacceptable response times; most conventional algorithms have $O(N^2)$ time complexity in the worst case scenario, where N is the number of nodes (intersections) in the road network. Routing algorithms with such a time complexity are impractical in map-based mobile applications where real-time optimal routes, some on large networks, are needed and where they typically feature mobile devices with limited CPU, memory, and power. Therefore, heuristic routing algorithms play a major role in map-based mobile applications and their proper design requires a thorough understating of their accuracy and time performance.

Much research in the past few decades has been focused on developing fast algorithms resulting mostly in heuristic routing algorithms that produce local-optimal solutions. For example, see Fu et al. (2006) for a survey of heuristic shortest path algorithms and Cherkassky et al. (1996) for an overview of theoretical and experimental studies of various shortest route algorithms. Any real-time routing algorithm can be of practical use in map-based mobile applications only when it produces reasonable solutions in an acceptable response time with real road network data. To date, there have been very few new routing algorithms that are tested for real-time processing using real road network data (Jacob et al. 1999; Jagadeesh et al. 2002; Kim et al. 2005a). A review of the differences between using real road network data and randomly generated network data along with the computational study of routing algorithms using realistic networks can be found in Jakob et al. (1999) and Liu (1997).

Chabini and Shan (2002) adapted A* algorithm to shortest path problems in dynamic deterministic networks. They evaluated the algorithm on the randomly generated network containing 3000 nodes and 10,000 links, and 100 time intervals. They compared the dynamically adapted A* algorithm and the dynamically adapted Dijkstra's algorithm and reported that the dynamically adapted A* algorithm resulted in a saving ratio of 11 in terms of nodes selected and a saving ratio of five in terms of computational times. Huang et al. (2006) extended Lifelong Planning A* algorithm to solve dynamic deterministic shortest path problems. They suggest the use of an ellipse to prune the unnecessary nodes to be searched and experimentally showed that the number of examined nodes could be 70-80% less than that of the A* algorithm. This corresponds to 17-31% savings in computational time needed to calculate shortest paths using test road networks in the experiments. Kim et al. (2005b) proposed to model the problem of computing dynamic stochastic shortest paths with traffic congestion information as discrete-time finite horizon Markov decision process. They developed decision-making procedures for determining optimal driver attendance times, optimal departure times, and optimal routing policies under time-varying traffic flows for just-in–time delivery services. They tested their method using real-time traffic congestion information for the road network in southeast Michigan. They achieved reduction in travel time of approximately 9.8216.19%. (Kim et al. 2005a) also improved efficiency of their previous algorithm.

Map-based mobile applications' effectiveness is often measured by the accuracy and time performance of the routing algorithms on which they are based. Jagadeesh et al. (2002) have suggested a routing approach that combines hierarchical and heuristic techniques based on road classification. They tested their approach using the road network of Singapore. On average, the routes computed by their approach were 3.31% longer than the shortest routes. Zhao and Weymouth (1991) proposed an adaptive route guidance technique that alternates between two different heuristic search algorithms based on the time available for route computation. Jung and Pramanik (2002), Chabini and Shan (2002), and Karimi (1996) developed a heuristic routing algorithm that limits the number of nodes used in computation by devising a window with two of its sides parallel with the straight line connecting the origin and destination nodes. However, while this window-based heuristic routing algorithm provides reasonable time performance, further research was required to realize accuracy and time performance of the algorithm based on a variety of possible windows (or subnetworks); windows can geometrically have different size and structure.

The work presented in this chapter is based on the window-based heuristic routing algorithm developed by Karimi (1996), and is focused on different approaches (window size and orientation) in limiting the search space that contains real road network data. The objective of this work was to test the hypothesis that the size and orientation of a subnetwork impact the accuracy and time performance of the window-based heuristic routing algorithm. To test this hypothesis, two subnetwork approaches (window size and orientation) were taken to reduce the search space in the window-based heuristic routing algorithm. The impacts of different sizes and orientations of windows on accuracy of the solutions and the time performance by the window-based heuristic routing algorithm were analyzed. Both approaches were tested using real road network data providing meaningful results applicable to real-world map-based mobile applications.

The main contributions of this chapter are: (a) analysis of different approaches for reducing the search space in the window-based heuristic routing algorithm and (b) testing of the window-based heuristic routing algorithm using real road network data. Realization of the impact of window size and orientation on accuracy and time performance by the window-based heuristic routing algorithm will help developing optimal solutions that meet the requirements of map-based mobile applications. The structure of the chapter is as follows. The window-based heuristic algorithm is described in section 12.1. In section 12.2 the experiments conducted are discussed. The results are discussed in section 12.3. Conclusions and ideas for future research are given in section 12.4.

12.2 Window-based heuristic algorithm

Since the approaches presented are based on Dijkstra's algorithm, the following brief description of the algorithm is given. A detailed description of Dijkstra's algorithm can be found in standard books on algorithms (e.g., Bertskas and Gallager 1992). Variations of Dijkstra's algorithm can be found in several publications, including Fredman and Tarjan (1987) and Gallo and Pallottino (1988). Dijkstra's algorithm computes the shortest route from a single, source, vertex to all other vertices in a weighted, directed network $G(N, A)$, where N is a finite set of nodes and A is a finite set of arcs. Associated with each $arc(i,j) \in A$ is its length (the cost is a function of length) $l_{i,j} \geq 0$. The time necessary to compute a shortest route is approximately a quadratic function of the network size, $O(N^2)$, where N is the number of nodes in the network. However, a time complexity of $O(A+NlogN)$, with A arcs and N nodes in the network, is possible by implementing a priority queue with a Fibonacci heap (Fredman & Tarjan, 1987).

The main idea behind the window-based heuristic algorithm is to limit the number of nodes used for computation by using a subnetwork of the original network (a window overlaid over the original network). The window-based heuristic algorithm first creates a subnetwork by constructing a rectangle that includes the origin and destination nodes and then applies Dijkstra's algorithm to the subnetwork. Since there are conceivably different approaches to devise such subnetworks, in this work we tested the following hypothesis. Subnetworks with different sizes and orientations will impact accuracy of the solutions and the time performance by the window-based heuristic algorithm. To test this hypothesis, we took two approaches to limit the search space and compared the solutions and the performance by the window-based heuristic algorithm. In the remainder of this section the two approaches are described.

12.2.1 Orientation-based window (OBW)

OBW was described by Karimi (1996) as a window oriented in the direction of the line connecting the origin and destination nodes. The window is constructed so that the origin and destination nodes lie on the long axis of symmetry of the rectangle. The size of an OBW is determined by the Euclidean distance, L, between the origin and destination nodes and by the parameters f_{sx} and f_{sy} which specify the percentage of increase in the size of the edge with respect to L, so that the former is for the longer side and the latter is for the shorter side. Therefore, the shorter side of the rectangle is $2Lf_{sy}$ and the longer side is $(2Lf_{sx} + L)$ as shown in Fig. 12.1

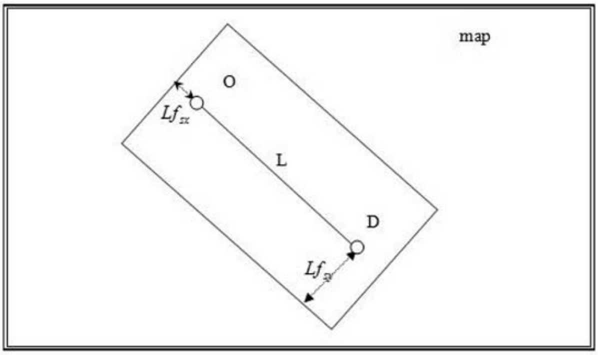


Fig.12.1. An example of OBW.

12.2.2 Parallel-based window (PBW)

PBW has sides parallel with the horizontal and vertical axes of the geographic extent (i.e., parallel to the x and y axes of the coordinate system in which the road network is presented) of the road network. PBW is constructed as depicted in Fig. 12.2. First, a rectangle whose sides are parallel with the x and y axes and the origin and destination nodes are on its diagonal is constructed. Each side of the rectangle is then increased by *2b* to construct a larger rectangle. PBW constructed in this way has a buffer *b* with the length of L+2b and the width of *2b*. A special case of PBW is when the abscissa connecting the origin and destination nodes lie on the vertical or horizontal axis.

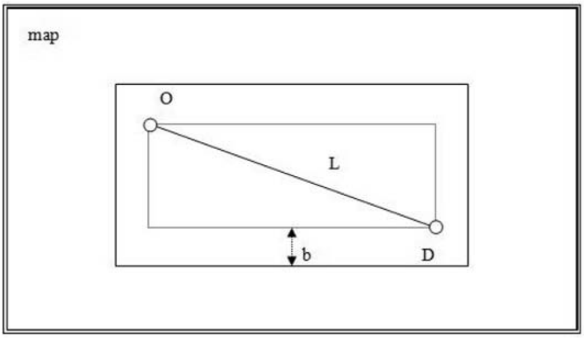


Fig. 12.2. An example of PBW

Several cases may occur when selecting links (road segments) for OBW and PBW. Example cases are illustrated in Fig. 12.3. Note that there are usually two types of roads in a road network database, the end nodes (intersection nodes) and the shape nodes (the nodes making the geometry of the road). While the intersection nodes are needed to provide the extent of a road segment, the shape nodes are needed to represent the geometry of the road segment. In Fig. 12.3, road segments a, b, c, and d are included and road segments e and f are not included in the sub-network.

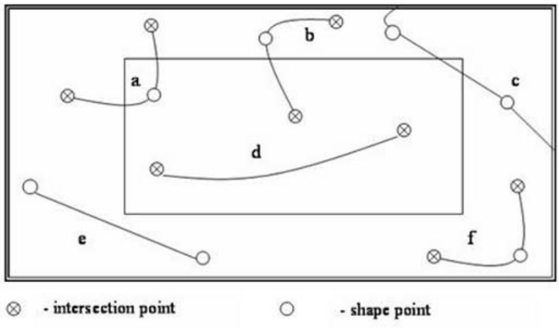


Fig. 12.3. Possible cases for OBW and PBW

12.3 Experiments

An empirical study was carried out to realize the solutions and performances of the two approaches (OBW and PBW) using a real road network data set. All tests were implemented on a 2GHz Pentium 4 with 500MB of memory. All program codes for simulations were written in Java. The road network data set used in the experiments was obtained from the Topologically Integrated Geographic Encoding and Referencing (TIGER) database (TIGER/Line Files 2002a; TIGER/Line Files 2002b); data from Record Type 1 (RT1) and Record Type 2 (RT2) of Allegheny County. RT1 files contain intersection nodes (end nodes) and RT2 files contain shape nodes. The road network used in the experiments consists of 59,861 nodes with a ratio of 1.33 between the links and the nodes. A total of 190 origin-destination (O-D) pairs were randomly selected. The Euclidean Distance (ED) between an O-D pair is one factor that determines the window (PBW or OBW) size and the number of nodes in the window.

Shortest routes generated by Dijkstra's algorithm on the road network were used as the baseline for performance and accuracy assessment. Twelve different window sizes were used (see Table 12.2 and Table 12.2). Note that for OBW (Table 12.1) the scale factor in both x and y directions are indicated while for PBW (Table 12.2) the buffer width is indicated. Dijkstra's algorithm yielded, for each computed route, the number of roads in the route, the route length, and the computation time. The average time and error (the difference between the shortest route using the entire road network and the shortest route using the subnetwork for a given origin-destination pair) rate for each window size using all computed routes were calculated. The computation time for each of the following was measured:

1. Creating the window (OBW or PBW)
2. Selecting the nodes and links for the window (OBW or PBW)
3. Creating the adjacency matrix
4. Running Dijkstra's algorithm

The accuracy of the results by PBW and OBW, which include incomplete routes and local-optimal routes, was assessed. An incomplete route is defined as one which misses one or more roads necessary to make up the route between the origin and destination points. A local-optimal route, using the subnetwork, is defined as a route that is longer than the shortest (optimal) route, using the entire network, between the same origin and destination points.

Table 12.2. Window sizes for OBW

Window number		1	2	3	4	5	6	7	8	9	10	11	12
Scale factor [%]	fsx	3%	5	7	9	11	13	15	17	19	21	23	25
	fsy	6%	10	14	18	22	26	30	34	38	42	46	50

Table 12.3. Window sizes for PBW

Window number		1	2	3	4	5	6	7	8	9	10	11	12
Buffer width [m]	b	150	250	500	1000	1500	2000	2500	3000	3500	4000	4500	5000

The number of nodes in the window also depends on the locations of the origin and destination points and the network density (number of nodes and links) within the window. Tables 12.3a and 12.3b show average number of nodes for different window sizes, ratios between number of links and number of nodes in each window, and basic statistics on the data. These ratios are from 0.93 to 1.47 for PBW and from 0.88 to 1.48 for OBW. The ratio between the numbers of links and nodes is low compared to the ratio of 2.56 presented by Jagadeesh et al. (2002) for the Singapore road network. Anderson et al. (1998) also analyzed road networks using TIGER/files and found the ratio between links and nodes as 1.38. The reason for the low ratio between the links and nodes in road networks from TIGER is that in addition to the intersections (end nodes), the shape nodes between the intersections (these are for representing the geometry of the roads which are stored in RT2) as well as the nodes that lie on the window border are considered.

Table 12.3a. Average number of nodes in a PBW, ratios between number of links and number of nodes, and basic statistics where NoR is the number of routes.

ED [km]			Window Size												NoR
			1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	
Short (< 5)	Nodes	Ave	620	1002	1417	1845	2280	2739	3223	3689	4165	4644	5143	5659	48
	Ratio Links/ Nodes	Ave	1.25	1.30	1.33	1.34	1.35	1.36	1.36	1.36	1.37	1.37	1.37	1.37	
		STD	0.12	0.11	0.11	0.10	0.09	0.09	0.09	0.09	0.09	0.09	0.09	0.08	
		Min	0.88	0.93	0.91	0.98	1.02	1.03	1.01	1.00	0.99	1.01	1.03	1.05	
		Max	1.41	1.45	1.45	1.45	1.47	1.48	1.47	1.48	1.48	1.48	1.48	1.47	
Low mid (5-10)	Nodes	Ave	2201	3609	5093	6582	8085	9643	11205	12763	14349	15930	17536	19134	73
	Ratio Links/ Nodes	Ave	1.32	1.34	1.35	1.36	1.36	1.36	1.37	1.37	1.37	1.37	1.37	1.37	
		STD	0.08	0.07	0.07	0.06	0.06	0.06	0.06	0.05	0.05	0.05	0.05	0.04	
		Min	1.09	1.11	1.15	1.17	1.16	1.17	1.18	1.19	1.19	1.20	1.21	1.21	
		Max	1.44	1.44	1.45	1.45	1.44	1.44	1.44	1.44	1.44	1.43	1.43	1.43	
High mid (10-15)	Nodes	Ave	4535	7354	10187	13021	15901	18749	21559	24304	26965	29472	31830	34129	52
	Ratio Links/ Nodes	Ave	1.32	1.33	1.34	1.35	1.35	1.35	1.35	1.35	1.35	1.35	1.35	1.35	
		STD	0.08	0.07	0.07	0.06	0.05	0.05	0.05	0.04	0.04	0.04	0.04	0.03	
		Min	1.07	1.14	1.15	1.17	1.17	1.17	1.17	1.17	1.17	1.17	1.18	1.19	
		Max	1.42	1.42	1.43	1.42	1.42	1.42	1.42	1.41	1.41	1.40	1.40	1.39	
Long (>15)	Nodes	Ave	8059	13166	18142	22771	27353	31571	35716	39372	42530	45147	47424	49362	17
	Ratio Links/ Nodes	Ave	1.35	1.36	1.36	1.37	1.37	1.36	1.36	1.36	1.36	1.35	1.35	1.35	
		STD	0.04	0.04	0.03	0.03	0.02	0.02	0.02	0.02	0.01	0.01	0.01	0.01	
		Min	1.23	1.25	1.26	1.29	1.30	1.31	1.32	1.33	1.33	1.33	1.33	1.33	
		Max	1.40	1.40	1.40	1.41	1.41	1.41	1.40	1.39	1.39	1.38	1.37	1.37	

Table 12.3b. Average number of nodes in an OBW, ratios between number of links and number of nodes, and basic statistics where NoR is the number of routes.

ED [km]			Window Size												NoR
			1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	
Short	Nodes	Ave	620	1002	1417	1845	2280	2739	3223	3689	4165	4644	5143	5659	48
	Ratio	Ave	1.25	1.30	1.33	1.34	1.35	1.36	1.36	1.36	1.37	1.37	1.37	1.37	
(< 5)	Links/	STD	0.12	0.11	0.11	0.10	0.09	0.09	0.09	0.09	0.09	0.09	0.09	0.08	
	Nodes	Min	0.88	0.93	0.91	0.98	1.02	1.03	1.01	1.00	0.99	1.01	1.03	1.05	
		Max	1.41	1.45	1.45	1.45	1.47	1.48	1.47	1.48	1.48	1.48	1.48	1.47	
Low mid	Nodes	Ave	2201	3609	5093	6582	8085	9643	11205	12763	14349	15930	17536	19134	73
	Ratio	Ave	1.32	1.34	1.35	1.36	1.36	1.36	1.37	1.37	1.37	1.37	1.37	1.37	
	Links/	STD	0.08	0.07	0.07	0.06	0.06	0.06	0.06	0.05	0.05	0.05	0.05	0.04	
(5-10)	Nodes	Min	1.09	1.11	1.15	1.17	1.16	1.17	1.18	1.19	1.19	1.20	1.21	1.21	
		Max	1.44	1.44	1.45	1.45	1.44	1.44	1.44	1.44	1.44	1.43	1.43	1.43	
High mid	Nodes	Ave	4535	7354	10187	13021	15901	18749	21559	24304	26965	29472	31830	34129	52
	Ratio	Ave	1.32	1.33	1.34	1.35	1.35	1.35	1.35	1.35	1.35	1.35	1.35	1.35	
	Links/	STD	0.08	0.07	0.07	0.06	0.05	0.05	0.05	0.04	0.04	0.04	0.04	0.03	
(10-15)	Nodes	Min	1.07	1.14	1.15	1.17	1.17	1.17	1.17	1.17	1.17	1.17	1.18	1.19	
		Max	1.42	1.42	1.43	1.42	1.42	1.42	1.42	1.41	1.41	1.40	1.40	1.39	
Long	Nodes	Ave	8059	13166	18142	22771	27353	31571	35716	39372	42530	45147	47424	49362	17
	Ratio	Ave	1.35	1.36	1.36	1.37	1.37	1.36	1.36	1.36	1.36	1.35	1.35	1.35	
(>15)	Links/	STD	0.04	0.04	0.03	0.03	0.02	0.02	0.02	0.02	0.01	0.01	0.01	0.01	
	Nodes	Min	1.23	1.25	1.26	1.29	1.30	1.31	1.32	1.33	1.33	1.33	1.33	1.33	
		Max	1.40	1.40	1.40	1.41	1.41	1.41	1.40	1.39	1.39	1.38	1.37	1.37	

12.4 Analysis of results

Shortest routes calculated by Dijkstra's algorithm using the entire network were used as a baseline to compare the two approaches. Of the 190 O-D pairs randomly selected, Dijkstra's algorithm was able to find routes between 188 O-D pairs. The reason why no routes were computed between two of the O-D pairs is that one of the origin or destination nodes in each pair was located on the county border, requiring road data from the adjacent county which is not part of the Allegheny County road network. The average time necessary to find the optimal routes between all 188 O-D pairs applying Dijkstra's algorithm to the entire network was 334.1 seconds with a standard deviation of 6.7 seconds.

Two types of errors were analyzed using OBW and PBW: (1) an incomplete route and (2) a local-optimal route. A route is incomplete when the algorithm could not connect the origin and destination nodes. A local-optimal route, using a subnetwork, occurs when the length of a route between the O-D pairs is longer than the optimal route using the entire network. The total error is defined as the sum of incomplete and local-optimal routes.

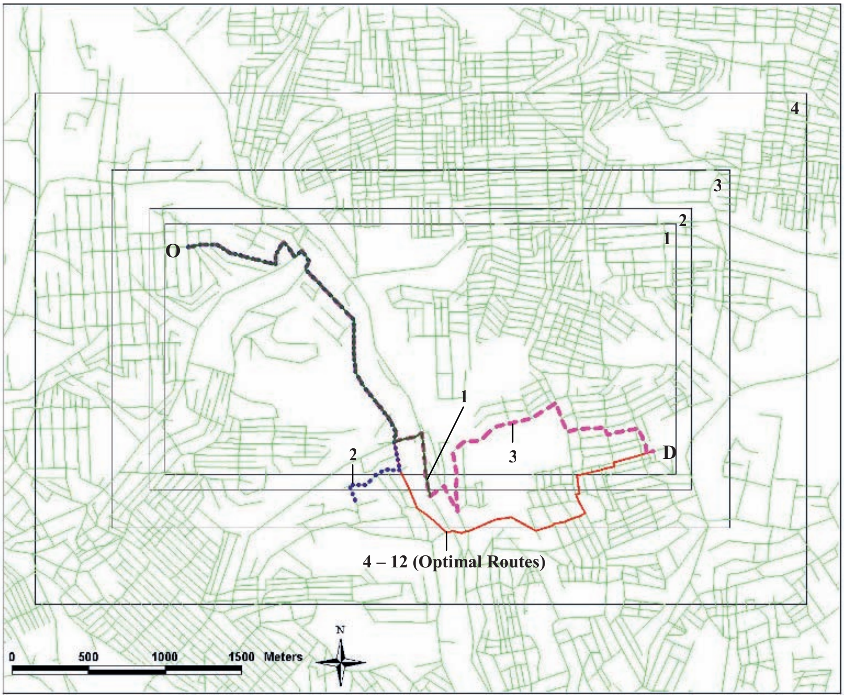


Fig.12.4. Computed routes between an O-D pair for different window sizes in PBW.

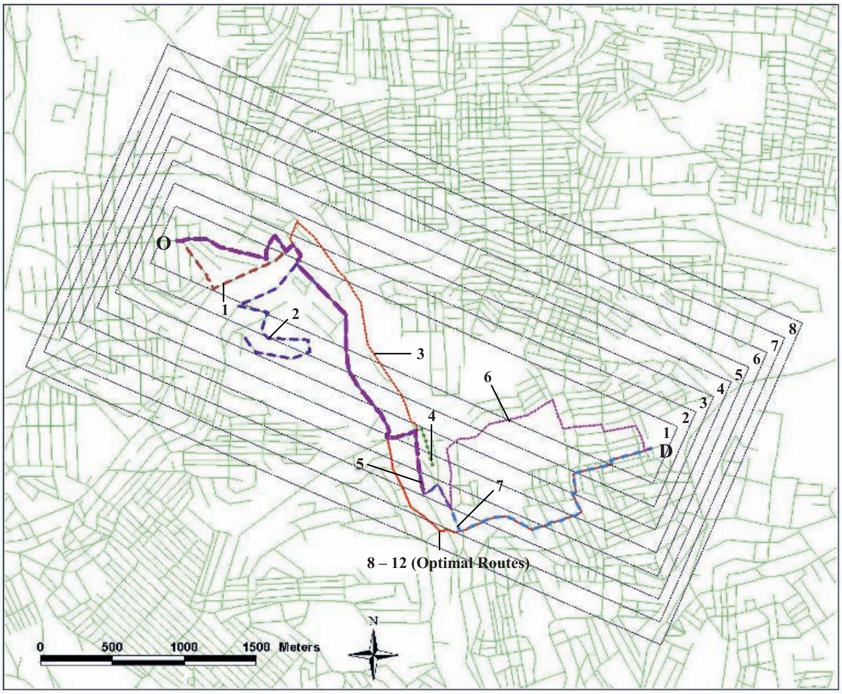


Fig.12.5. Computed routes between an O-D pair for different window sizes in OBW.

Fig. 12.4 and 12.5 show examples of incomplete routes (in small windows), local-optimal routes, and optimal routes for PBW and OBW, respectively. Fig. 12.4 shows the results of PBWs: incomplete routes for windows 1 and 2, local-optimal routes for window 3, optimal routes for the other windows (4-12). Fig. 12.5 shows the results of OBWs: incomplete routes for windows 2, 3 and 5, local-optimal routes for windows 6 and 7, and optimal routes for the other windows.

Table 12.4. Basic statistics for route lengths and ED ratios.

Method	Average	STD	Min	Max
Dijkstra	1.22	0.12	1.05	1.87
PBW	1.25	0.11	1.05	1.87
OBW	1.24	0.11	1.05	2.28

Table 12.4 shows the basic statistics of route length and ED ratios. The average ratio for PBW and OBW, 1.25 and 1.24, respectively, is slightly higher than for the baseline route calculated for the entire network (1.22). This is because the

local-optimal routes were used to compute the route lengths and the ED ratios for PBW and OBW.

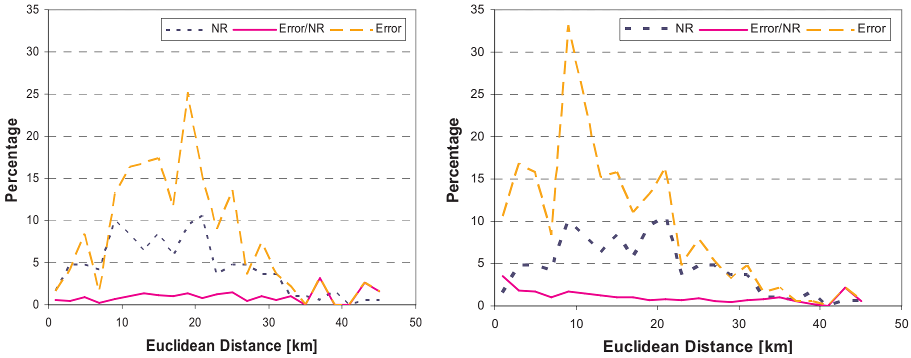
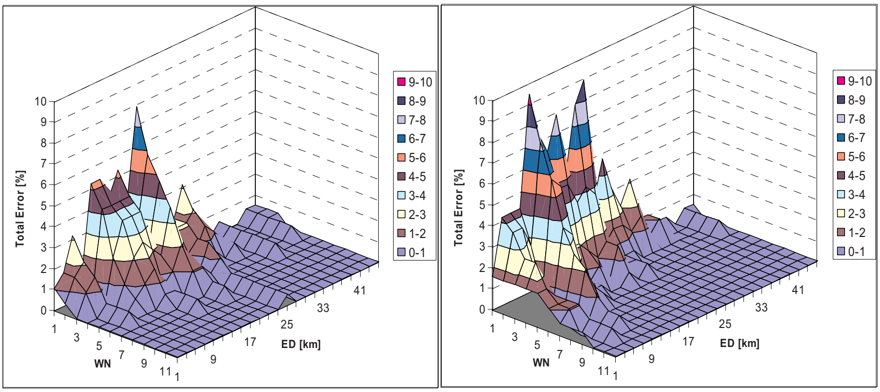


Fig.12.6. The number of errors depends on window size and ED: (a) PBW and (b) OBW.



(a) (b)

Fig. 12.7. Distributions of percentage of routes (NR), percentage of total error (Error), and percentage of error divided by number of routes (Error/NR) based on EDs: (a) PBW and b) OBW.

As expected, smaller windows produced larger errors. Fig. 12.6 shows that the total error distribution depends on the window size and the ED between the origin and destination nodes. The errors are more concentrated in the middle of ED scale for PBW (see Fig. 12.6a). In Fig. 12.6b, it is shown that the errors are more probable for shortest EDs. The selected routes were grouped into equidistant intervals with a length of 2 km based on the ED. Fig. 12.7 shows the distribution of the total error and the distribution of error rates (the distribution of errors divided by the distribution of EDs). EDs were not uniformly generated. More O-D pairs with an ED in the range between 8 km and 24 km were generated which can partially explain the distribution of the total error. The distribution of errors follows the distribution of EDs for PBW (see Fig. 12.7a). This means that errors are generated homogeneously independent from EDs. The distribution of total errors for OBW is skewed to the smallest ED and the error rate is descending as a function of ED (see Fig. 12.7b). The result of the experiments showed that errors are more probable for smallest windows than for larger windows using OBW.

Fig. 12.8 and Table 12.5 show percentages of incomplete and local-optimal routes for each window size used in PBW and OBW. The percentage of errors was calculated for each window size separately using the number of calculated routes (188) in the window. As expected, more errors occurred for smaller windows, especially using OBW where both errors are larger than those for PBW, but they drop faster. The experiment showed that the number of incomplete and local-optimal errors in OBW is 20% higher than PBW for all window sizes.

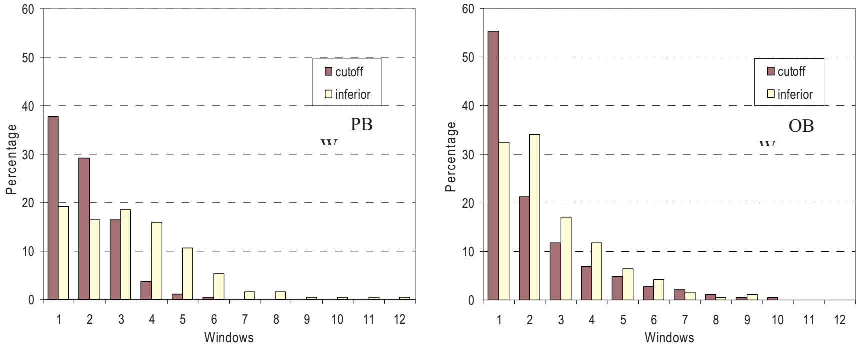


Fig.12.8. Percentage of incomplete and local-optimal routes for each window size

Table 12.5. Percentage of errors and performance.

Window number	PBW				OBW			
	Percentage of errors [%]			Performance	Percentage of errors [%]			Performance
	Incomplete	inferior	total	Times [s]	Incomplete	inferior	total	Times [s]
1	37.8	19.2	57.0	11.71	55.3	32.5	87.8	2.10
2	29.3	16.5	45.8	12.13	21.3	34.0	55.3	4.71
3	16.5	18.6	35.1	14.34	11.7	17.0	28.7	8.51
4	3.7	16.0	19.7	17.35	6.9	11.7	18.6	13.43
5	1.1	10.6	11.7	20.95	4.8	6.4	11.2	19.59
6	0.5	5.3	5.8	25.1	2.7	4.3	6.9	26.63
7		1.6	1.6	29.71	2.1	1.6	3.7	34.67
8		1.6	1.6	34.61	1.1	0.5	1.6	43.01
9		0.5	0.5	40.05	0.5	1.1	1.6	51.52
10		0.5	0.5	45.89	0.5		0.5	60.76
11		0.5	0.5	52.43				69.16
12		0.5	0.5	57.02				78.30

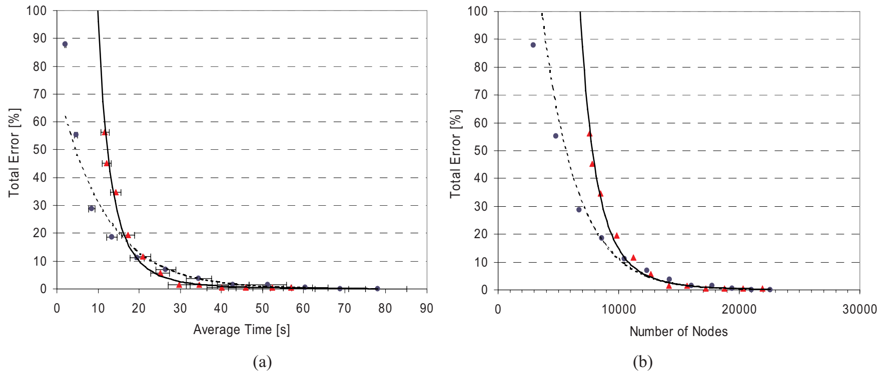


Fig.12.9. (a) Performance versus error and (b) dependence of relative total errors on the number of nodes in windov (dashed line – OBW; solid line – PBW).

Fig. 12.9a and 12.9b present the dependence of the total error for each window size on the average time (performance) and the average number of nodes, respectively. From these figures and Table 12.5, which summarizes the values of errors and average performances for all window sizes used for PBW and OBW, it can be concluded that for smaller windows, the performance of OBW is better than the performance of PBW, but the error in this region is higher than 20%. When the error is small there is no statistically significant difference to prefer OBW or PBW to reduce the search space in the window-based heuristic algorithm to compute optimal routes in real time.

Fig. 12.10 shows the distribution of local-optimal and optimal routes ratios. It can be seen that over 50% of the local-optimal routes are less than 3% longer than the optimal routes and about 90% of them are not longer than 10% of the optimal routes. The average ratio of the lengths of local-optimal and optimal routes is 1.046 and 1.042 for OBW and PBW, respectively.

Since the performances of both OBW and PBW partially depend on the underlying road network size and density, an analysis of road networks is given. A typical road network is irregularly distributed and unequally developed in the various parts of any geographic extent. This can be caused by the terrain or settlement in some regions. Although the highest road network density is expected in urban areas, there can be heterogeneity in road distribution, especially in the areas between neighborhoods, which are usually divided by terrain obstacles such as hills (parks and forests) and rivers (where the density of bridges is important). The road network of Allegheny County has similar characteristics and its density is high in the center of the network (downtown area) and gradually decreases from the center to the boundary lines. The density of a road network has no impact on the outcomes when Dijkstra's algorithm is applied to the entire network. However, the density of a road network impacts the accuracy and performance of OBW and PBW. For example, when a small window is chosen for the purpose of improving performance, a route may not be found and if a route is found the accuracy may not be high.

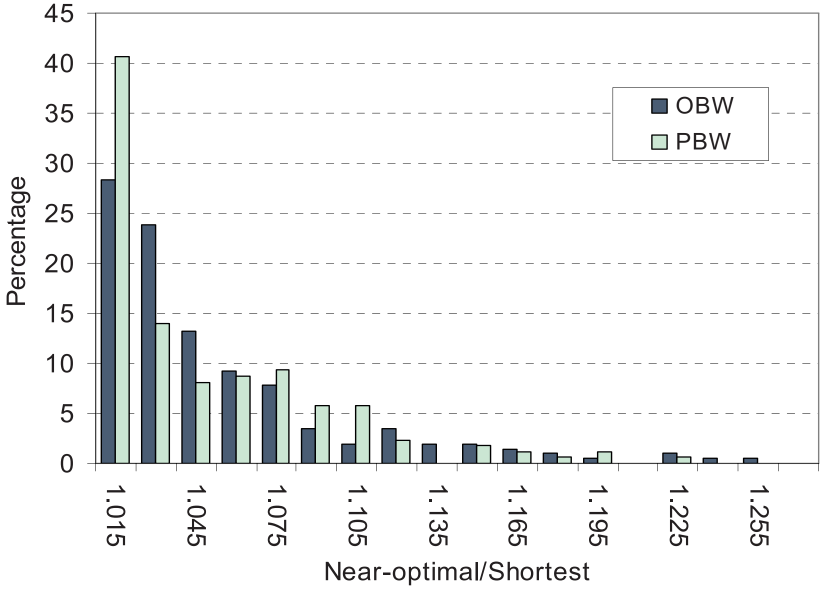


Fig.12.10. Local-optimal and optimal route lengths ratio distribution for OBW and PBW.

Knowledge about a road density can be employed in the window-based heuristic algorithm as one of the factors determining optimal window sizes. Generally, the optimal window size can be specified as $s(x,y) = f(\rho, t, \zeta, \varepsilon)$ where ρ is the density of road network, t is the computing time, ε is an error function and ζ is the degree of connectivity defined as the number of links to the number of nodes ratio. This means that an optimal window size is the smallest area (subnetwork) of the original road network where acceptable solutions and performances can be obtained.

This idea was used in an example to show how knowledge about networks could improve performance of the window-based heuristic algorithm (see Fig. 12.11). The information about the network density is presented by $\rho(x,y)$, where x, y coordinates are used to specify conditionally the width of the window. Let abscissa AB that passes through O-D pairs be described by equation $ax+b$ where its length is $l = 1.25|OD|$. First, we divide abscissa l into η intervals each with $\Delta L = \frac{l}{\eta}$ length. Then we divide the road network into regions by lines that are perpendicular to the abscissa. For each interval $[i, i+1]$, where point i has coordinates (x_i, y_i) on the boundary of the interval, we can specify the distance as $\sqrt{(y_2(x_{i+1}+h,h) - y_1(x_{i+1},h))^2 + h^2}$ from abscissa l that will specify the area from which the nodes used for a given interval using the following inequality:

$$\int_{x_i}^{x_{i+1}+h} \int_{y_1(x,h)}^{y_2(x,h)} \rho(x,y)\,dxdy \geq \gamma$$

where γ is a parameter that specifies the number of links that are optimal for a given interval, ΔL. From the equation above we can specify h and the last edge of the window (bold in Fig. 12.11). The windows for each interval are determined in the same way. This example shows how the relevant nodes can be selected more accurately when knowledge about the network is used. Future work could include testing these methods on different road networks.

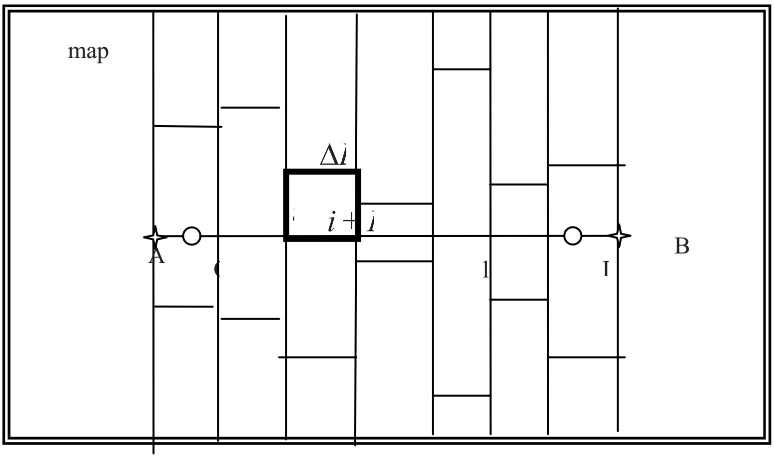


Fig.12.11. Example of selecting intervals and width of the window.

12.5 Conclusions and future research

Two approaches (OBW and PBW) based on a window-based heuristic algorithm to find shortest routes between given O-D pairs were presented. It was empirically shown that OBW and PBW reduce the average time of computation by a factor between 5 and 87. This means that they are potential for real-time computation of shortest routes in very dense networks. The analysis of the results by OBW and PBW showed that there is not a statistically significant difference between them. However, with a subnetwork of less than 10,000 nodes (with an average computing time of less than 15 seconds), OBW has better performance and accuracy than PBW and for a subnetwork with more than 10,000 nodes the difference between the two approaches is not significant. This indicates that there are more nodes in small OBWs than in small PBWs that are part of the optimal solution. The same does not hold for large windows, as the size of the window increases the influence of its orientation diminishes. This is because in both approaches the geographic extent is a factor determining the number of nodes in windows. Another issue is the number of nodes that are part of optimal solutions. The proportion of the nodes that are part of optimal solutions to the nodes that are not part of optimal solutions is higher for smaller windows and lower for larger windows. For small windows a higher accuracy change at the expense of a lower performance change is achieved, and the opposite is true for large windows as can be seen in Fig. 12.9(a).

The performance and accuracy of OBW and PBW depend on the location of the origin and destination points, the ED between them and their orientation, and the road network density. If the origin or the destination point is located close to the border of an open road network (selected from a larger network), a route between this point and any other location in the network can not be computed as the connection to this point is through a neighboring county. This problem can be overcome by taking into account all possible routes including those in the neighboring counties between all the points near or on the boundary of the road network.

One area for future research is determination of optimal window sizes to eliminate incomplete routes. The results of the experiments showed that the elimination of incomplete routes and the acceptance of local-optimal routes lead to routes which may be 4.6% longer on average than optimal routes (those computed by using the entire network). Another area for future research is the utilization of other criteria than shortest distance, for example travel time, road type, road difficulties, and traffic conditions (Bovy and Stern 1990; Lotan 1997; Pang et al. 1999). The proposed method can be naturally extended to constrained shortest path problems when the shortest distance criteria are combined with other conditions such as road type, a-autonomy shortest distance (Terrovitis et al. 2005), and curvature of the road segments. This method of limiting search space by OBW and PBW can also be applied to the problems of stochastic and dynamic shortest routes.

The work presented here also provided an insight into finding a balance between performance and accuracy. This insight can be used in future to design systems that will be able to choose parameters of the window based on user's preferences and knowledge of the specific road network.

References

Andersson, G., Francis, R. L., Normark, T., and Rayco, M. B. (1998): "Aggregation method experimentation for large-scale network location problems." *Location Science*, 6, 25-39.

Bertskas, D. P., and Gallager, R. (1992): *Data Networks*, Englwood Cliffs, NJ: Prentice Hall.

Bovy, P. H. L., and Stern, E. (1990): *Route Choice: Wayfinding in Transport Networks*, Kluwer Academic Publishers, Boston, MA.

Chabini, I., and Shan, L. (2002): "Adaptations of the A* algorithm for the computation of fastest paths in deterministic discrete-time dynamic networks." *Intelligent Transportation Systems, IEEE Transactions on*, 3(1), 60-74.

Cherkassky, B. V., Goldberg, A. V., and Radzik, T. (1996): "Shortest paths algorithms: theory and experimental evaluation." *Math. Program.*, 73(2), 129-174.

Fredman, M. L., and Tarjan, R. E. (1987): "Fibonacci heaps and their uses in improved network optimization algorithms." *Journal of the ACM*, 34(3), 596-615.

Fu, L., Sun, D., and Rilett, L. R. (2006): "Heuristic shortest path algorithms for transportation applications: State of the art." *Computers & Operations Research Part Special Issue: Operations Research and Data Mining*, 33(11), 3324-3343.

Gallo, G., and Pallottino, S. (1988): "Shortest path algorithms." *Annals of Operations Research*, V13(1), 1-79.

Huang, B., Wu, Q., and Zhan, F. B. (2006): "A Shortest Path Algorithm with Novel Heuristics for Dynamic Transportation Networks." *International Journal of Geographical Information Science*.

Jacob, R., Marathe, M. V., and Nagel, K. (1999): "A computational study of routing algorithms for realistic transportation networks." *ACM Journal of Experimental Algorithmics.* vol.4; 1999.

Jagadeesh, G. R., Srikanthan, T., and Quek, K. H. (2002): "Combining hierarchical and heuristic techniques for high-speed route computing on road networks." *Computing and Control Engineering Journal*, June 2002, 120-126.

Jung, S., and Pramanik, S. (2002): "An Efficient Path Computation Model for Hierarchically Structured Topographical Road Maps." *IEEE Transactions on Knowledge and Data Engineering*, 14(5), 1029-1046.

Karimi, H. A. (1996): "Real-time optimal-route computation: A heuristic approach." *ITS Journal*, 3(2), 111-127.

Kim, S., Lewis, M. E., and White, C. C. (2005a): "State Space Reduction for Nonstationary Stochastic Shortest Path Problems With Real-Time Traffic Information." *Intelligent Transportation Systems*, 6(3), 273 - 284.

Kim, S., Lewis, M. E., and White, C. C., III. (2005b): "Optimal vehicle routing with real-time traffic information." *Intelligent Transportation Systems, IEEE Transactions on*, 6(2), 178-188.

Liu, B. (1997): "Route finding by using knowledge about the road network." *IEEE Transaction on Systems , Man, and Cybernetics-Part A: Systemes and Humans*, 27(4), 436-448.

Lotan, T. (1997): "Effects of familiarity on route choice behavior in the presence of information." *Transportation Research Part C: Emerging Technologies*, 5(3), 225-243.

Pang, G. K. H., Takabashi, K., Yokota, T., and Takenaga, H. (1999): "Adaptive route selection for dynamic route guidance system based on fuzzy-neural approaches." *Vehicular Technology, IEEE Transactions on*, 48(6), 2028-2041.

Terrovitis, M., Bakiras, S., Papadias, D., and Mouratidis, K. (2005): "Constrained Shortest Path Computation." Advances in Spatial and Temporal Databases, 9th International Symposium, SSTD, Angra dos Reis, Brazil, 181-199.

TIGER/Line Files. (2002a): "UA Census 2000 TIGER/Line Files Technical Documentation. <http://www.census.gov/geo/www/tiger/tigerua/ua_tgr2k.html>." The U.S. Census Bureau, Washington DC.

TIGER/Line Files. (2002b):. "UA Census 2000 TIGER/Line Files: machine-readable data files. <http://www.census.gov/geo/www/tiger/tigerua/ua_tgr2k.html>." The U.S. Census Bureau, Washington DC.

Zhao, Y., and Weymouth, T. E. (1991): "An adaptive route guidance algorithm for intelligent vehicle highway system." *American Control Conference*, 3, 2568-2573.

13 How Mobile Maps Cooperate with Existing Navigational Infrastructure

Derek REILLY, Bonnie MACKAY, Kori INKPEN

Faculty of Computer Science, Dalhousie University

Abstract. Mobile map design often reflects contextual concerns like user population, activity and location, while ignoring tools in the environment that facilitate the same activities to be supported by the new application. However, it is rarely evident from the outset if a mobile map application should replace or augment existing tools. Augmenting tools like kiosk maps and signage may result in an application that is easier and more beneficial to use. This chapter presents a naturalistic study examining how mobile map interfaces can be used to augment other tools such as kiosk maps when navigating in buildings. The results of the study provide evidence that using kiosk maps alongside a mobile wayfinding application can promote the acquisition of spatial knowledge. Study participants were better able to trace their route and place key landmarks on a map when they had interacted with kiosk maps during wayfinding tasks. In addition, many participants actively integrated environmental cues beyond those explicitly referred to in the mobile application.

13.1 Introduction

Mobile map design places particular emphasis on context of use. Designers consider aspects of the region a map covers (e.g. is it landmark-rich, is the road layout regular or irregular), and the expected activity of map users (e.g. driving to the nearest bank or siteseeing on foot). A key element of context that is sometimes overlooked is the existing public navigational infrastructure (e.g. kiosk maps and signage). Even when not directly incorporated in a mobile map design, designers should be aware of how users will synthesize information from these various sources, and consequently how their applications will be used in practice. We therefore distinguish between environmental information that an application designer intends the user to make use of (e.g. landmarks described in route descriptions), and environmental information that is not explicitly designed for (e.g. signage, pedestrian flow, or the presence of concierges and info desks). When navigational aids already exist in an environment, it may be more effective to design mobile map services to augment rather than replace these aids.

In this chapter, we present work considering how personal mobile devices can be used alongside large, public, stationary maps or public kiosks when navigating an unfamiliar setting. We use the term ‘kiosk map’ here to refer to all such large public map displays. We focus on interactions between the mobile device and kiosk maps, and the relationship between the kiosk map’s presentation and the presentation on the mobile device. Previous work has shown that users are receptive to using personal devices to interact directly with large maps (Reilly, Dearman Welsman-Dinelle and Inkpen, 2005), and has suggested that this kind of interaction may promote the implicit acquisition of survey knowledge (Reilly, Rodgers, Argue, Nunes, and Inkpen, 2006). Further, cues in a mobile device’s presentation may elicit recall of salient aspects of the large map (MacKay, Watters and Duffy, 2004). An effective combination of mobile devices and kiosk maps should draw on the advantages of large displays while addressing some of the challenges of using maps on small devices, including limited screen space and poor interactivity. This configuration – combining mobile devices with static kiosk maps – is analyzed in a naturalistic setting in the study presented in this paper.

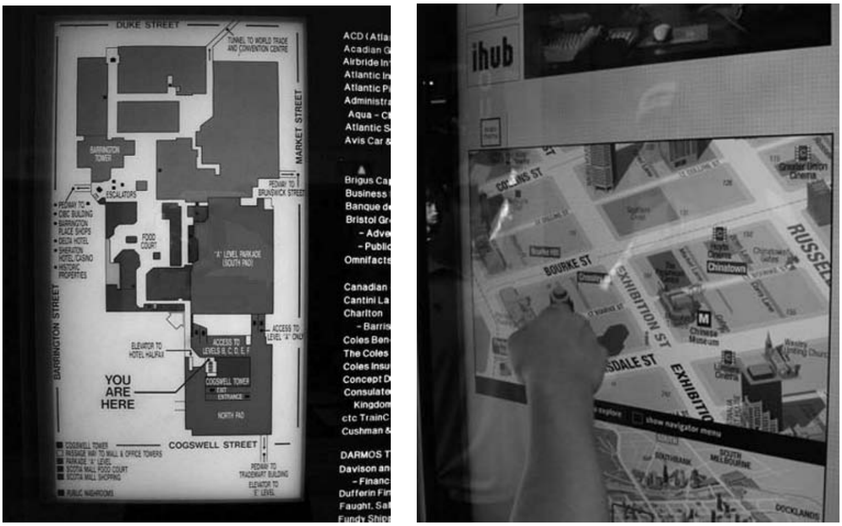
The study considers how route information on personal mobile devices can be used alongside kiosk maps when navigating in an unfamiliar setting. We focus on interactions between the mobile device and kiosk maps, and the relationship between the kiosk map’s presentation and the presentation on the mobile device. We observe the impact of interactivity and presentation on satisfaction, task completion, and acquired survey and route knowledge. This is accomplished by varying the level of interactivity between the mobile device and the kiosk maps, and varying the presentation style of the route information on the mobile device. We also observe more generally how study participants utilize environmental cues beyond those referenced by the mobile map application.

13.2 Background and motivation

13.2.1 Public kiosks

Public kiosks can provide static information (such as an area map, a historical vignette, or a building directory), or can be interactive, permitting transactions such as renewing a driver’s license, checking in for a flight, or paying for parking (see examples in Fig. 13.1). People expect kiosks to be available at appropriate locations (in a building lobby or beside an information booth in a shopping mall). Given a clear, well-designed kiosk map, people can recognize salient features at a glance and gather a rich array of information quickly (Peterson, 2004).

A feature of most interactive kiosks is that users can only perform simple, directed tasks (Johnston and Bangalore, 2004). For example, a tourist kiosk may show routes or sites selected from a small set of options. Kiosks tend to have limited input capabilities, usually providing a touch screen interface that takes up considerable screen real estate. Kiosk interfaces, while designed for public use, tend to support single-user interaction only.



(a) (b)

Fig. 13.1. (a) A non-interactive kiosk map displaying part of the setting used in the study presented in this chapter. (b) The ihub interactive kiosk map in Melbourne, Australia.

Some kiosks have incorporated other modes of interaction, such as speech (Johnston and Bangalore, 2004; Christian and Avery, 2000). However, there are problems with speech interfaces, such as failure to recognize commands and background noise interference. In cases where information is provided verbally, users can feel more like passive observers (Nivala and Sarjakoski, 2003), which may not suit the purpose of the kiosk. Other systems have incorporated multi-modal interfaces, such as MATCHKiosk (Johnston and Bangalore, 2004), GAUDI (Kray, Kortuem and Krüger, 2005) and Kimono (Huang, Pulli and Rudolph, 2005). With MATCHKiosk users interact with a city guide using speech, hand-writing, touch or with a set of multi-modal interactions. The resulting guide response is then printed out for the user to take away. The GAUDI Display System (Kray et al., 2005) combines a number of public kiosk displays situated across a university campus to help guide visitors, students and staff around campus for on-campus events. It provides dynamic signage for event-based navigation (for example, concerts or public lectures): when users pass a kiosk while navigating, the kiosk provides directional cues for their destination automatically. Kimono (Huang et al., 2005) (Kiosk-Mobile Phone Knowledge Sharing System) consists of several kiosks available in university campus buildings that provide immediate access to digital information (such as upcoming events and campus maps). Users interact with the system by touch, keyboard, speech and gesture. Users can then have items of interest transferred to their handset.

In previous work, Reilly et al. (2005, 2006) have considered ways in which mobile devices can be used alongside large static maps by permitting direct interaction using the mobile device as a pointer or lens. In the Marked-up Maps prototypes as shown in Fig. 13.2, a paper map is connected to a database of related information via RFID (radio frequency identification) tags, which are placed in a grid-like fashion on the back of the paper map. The database can then be queried using a mobile device to interact with the map.

Fig. 13.2. A Marked-up map with PDA interaction. In this prototype RFID tags are placed in a regular grid on the other side of the paper map, and a reader on the PDA permits the user to select and interact with map regions.

This model presents a number of potential benefits, including:

- Maintaining a large, carefully designed, static map while introducing the benefits of mobile computing (e.g. personalised, dynamic content).
- Allowing a public kiosk to become interactive while not constraining that interaction to a single user.
- Permitting a range of queries to be expressed visually and gesturally, reducing the amount of interaction using the mobile device alone.

A prior study evaluated the technique's ability to relate single points on a map to relevant information resources. The technique was compared against using a paper map with a standard electronic guide with an index. Results were promising for the technique, providing greater success during tasks that required a close correspondence between the map information and the guide resources (e.g. spatially-oriented tasks) (Reilly et al., 2006). The technique was also shown to significantly increase the amount of time spent using the paper map, while not increasing overall task time. In the study presented in this paper we examine in part how an application that uses this technique might promote the acquisition of spatial knowledge by encouraging interaction with a kiosk map. Hommel and Knuff (2000) provide evidence that objects can be conceptually grouped by the action performed on (or within) them, and that this makes spatial relations between these objects easier to judge. However, this was only shown to occur when the actions contained some implicit spatial relationship with the objects or were otherwise 'meaningful' for the objects. It is unclear whether direct interaction with symbols or other features on a paper map will impact spatial coding of these map features in the same way.

13.2.2 Maps on handheld devices

In many cases the information displayed on a kiosk is stationary – that is, it cannot go with an individual unless they take notes. Mobile maps can be viewed as a mobile kiosk that provides useful information at major decision points (Borchers, Deussen and Knörzer, 1995). The resources of mobile systems are more restrictive than their larger counterparts. Mobile devices have a small display area, cumbersome input modalities, limited bandwidth, communication costs, and energy consumption issues (Nivala and Sarjakoski, 2003; Reichenbacher, 2003; Zipf, 2002). Maps for mobile devices have to consider these limitations and restrictions.

The design parameters for mobile maps are quite different than those for large kiosk maps. While kiosk maps must reflect their broad setting and a range of purposes, mobile maps often represent a snapshot of an immediate information need (such as the route to a restaurant) (Kray, Laakso, Elting and Coors, 2003; Meng, 2005). There are many contextual factors that must be taken into account in mobile map design, including the users themselves, their task, their location, and the available technical resources (Bieber and Giersich 2001; Nivala and Sarjakoski, 2003; Kray and Baus, 2001; Zipf, 2002). This includes factors such as time of day (Bieber and Giersich 2001) and whether the user is in a hurry and needs to make quick decisions (Borchers et al., 1995; Reichenbacher, 2003). As with all maps, user demographics can influence both design and "lexical interpretation", as can personal preferences, visual literacy, domain knowledge and computer experience (Meng, 2005; Nivala and Sarjakoski, 2003; Zipf, 2002; MacEachren, 1995). Basic spatial cognitive ability and other cognitive aspects such as short term memory, particularly for mobile map users or kiosk visitors who will view the map only intermittently, also informs and constrains the presentation of information on the display (Meng, 2005). Users can experience a higher cognitive load using mobile devices, especially while navigating, as they must process competing visual stimuli on the device and in their environment simultaneously. If the "user" is actually more than one person, this can impact how mobile map services are designed and used (Reichenbacher, 2003). For example, one person might focus on the surroundings while another focuses on the device application.

Map-related task categories include navigation, recognizing or identifying a location, exploration, and understanding the spatial relationship between locations (Oulasvirta, Nivala, Tikka, Liikkanen and Nurminen, 2005; Nivala and Sarjakoski, 2003). A single task may be broken into subtasks of various types (for example, "turn left at x" involves a recognition step followed by a navigation step) (Tamminen, Oulasvirta, Toiskallio and Kankainen, 2004). In order to accomplish these tasks effectively using mobile map applications, users must "…construct an adequate mental representation of the … mapping between digital and physical space…" (Oulasvirta et al., 2005). People can also employ a variety of environmental cues (such as landmarks, intersections, landscapes, districts and areas, distance, topologies, etc) to help build an internal model of their environment (Lynch, 1960). Information gained from such cues is useful when conducting these tasks, with or without the additional support of a mobile map application.

Effective route maps must provide information that is necessary and sufficient to take appropriate key actions, such as choosing among several possible paths in a route (Daniel and Denis, 2004). While reducing visual clutter in map presentations can enhance map comprehension in general (Phillips and Noyes, 1982), it is particularly important for mobile map applications that route information is simplified and extra information is removed (Agrawala and Stolte, 2001). There are different techniques to help generalize route information for mobile devices (Agrawala and Stolte 2001; Zipf, 2002), such as applying the principle of readability (draw important objects larger and in a different scale), removing clutter, summarizing, distortion, simplification, and abstraction. Several studies (Look, Kottahachchi, Laddaga and Shrobe, 2005; May, Ross, Bayer and Tarkiainen, 2003; Tom and Denis, 2003) have demonstrated that providing landmarks in route descriptons is better than distance information or road names. The landmarks added to route information should have a distinct feature that is visible and can be clearly described (Sefelin, et al., 2005; Oulasvirta, et al., 2005).

Cues in a mobile presentation may elicit recall of salient aspects of a kiosk map. Research has shown that people have the ability after viewing an information space to recall and recognize that space even when distorted or shrunk to unreadable or illegible sizes. MacKay et al., (2004) looked at how to present a web page originally viewed on a large screen on a small device to support various tasks. Participants were more successful locating previously seen items in web pages when the look and layout of the original web page was maintained (either shrunken to fit within the screen or to scale with scrolling) than when changing the layout into a linear, textual list of items. Czerwinski, van Dantzich, Robertson and Hoffman (1999) reported that users, even after long periods of inactivity with certain web pages, could recall and retrieve particular pages from thumbnail representations. Kaasten, Greenberg, and Edwards (2001) also found that thumbnails promoted recognition, and identified several cues (colour, layout and images) important to recognition.

These results could also be applicable to maps first viewed on a kiosk station and then transferred to a mobile device. In the research presented in this chapter, we consider several route presentation formats, two of which are reflective of the kiosk map, and the third a textual list of steps in the route.

13.2.3 Signage and other environmental variables

The influence of signage and environmental variables other than landmarks (as traditionally defined) on navigation is well established. Weismen (1981) analysed wayfinding in buildings. He defined 4 broad categories of environmental variables that can impact wayfinding:

1. the presence of signage
2. the ability to see familiar cues or landmarks from a novel location
3. the extent to which one location looks different from another
4. the overall plan or layout of a setting

O'Neil (1991) emphasized the need to examine multiple environmental variables at once to understand how they impact wayfinding. He considered the relationship between floor plan complexity and various signage conditions, finding that graphical signage yielded the fastest travel times, while textual signage lead to fewer wayfinding errors. He also found that signage was only a moderate countermeasure to complex layouts.

While the work presented in this chapter considers the relationship between mobile maps and kiosk maps primarily, it was impossible to ignore the impact of other environmental factors in a naturalistic setting. As such we reflect also on the impact of factors such as signage and overall layout in our results.

13.4 Contextual design and experimental setting

In this section we outline our experiences assessing the experimental setting and designing the prototype interfaces used in our experiment. While the design decisions made were primarily influenced by the aim of the experiment, we still reflect on the availability of navigational infrastructure such as kiosk maps.

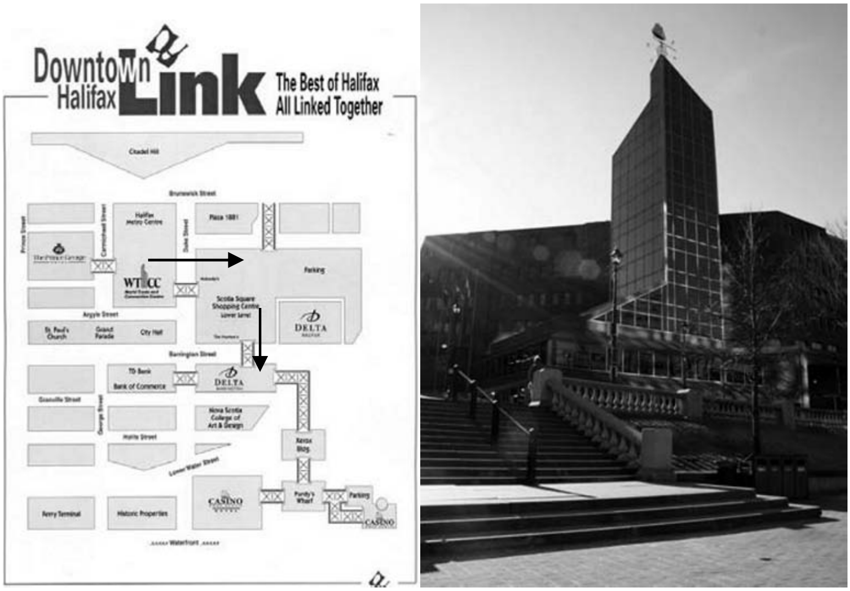
We considered several buildings as candidate locations for our experiment, by running through sample wayfinding scenarios in them. Not all buildings benefit from route information to the same extent. For example, an academic building we considered had a consistent layout and numbering scheme on each floor as well as a directory in the lobby, making route information less useful in most cases. Similarly, a campus library provided ample support for locating books (the usual task), although support for locating special collections, meeting rooms and service desks was less obvious. We tried to identify buildings that had an irregular layout and/or limited navigational support in the way of maps, directories or signage. It was decided to use a pedway (indoor walkway) system in the downtown core of the city of Halifax, Nova Scotia. The pedway joins several hotels, small shopping malls, business offices and a convention centre. We chose to use this location for three main reasons:

1. The pedway offers an indoor path between buildings, avoiding bad weather and city intersections.
2. The connected buildings have different functions and layouts, giving an interesting combined space in a small area.

3. The location was unfamiliar to many in the university community from which our sample was largely drawn.

Tasks for the study were designed around the theme of attending a conference being held in the World Trade and Convention Centre (WTCC). The tasks involved finding rooms in the conference centre itself, making a detour to your hotel, and finding travel agents in the adjacent shopping malls. Researchers walked through and revised each of the tasks several times, noting landmarks, signage, building layout and navigation decision points en route.

There is no kiosk map in the main lobby of the convention centre. Instead, the concierge provides a paper map if requested (Fig. 13.3). All four levels of the centre are presented on the same copy and are somewhat difficult to distinguish from each other (i.e. each level has a name, and not a level number associated with it). There was no indication on the map that the convention centre was integrated with an arena and a small office tower. There was no navigation signage in the lobby, aside from a list of events and their corresponding meeting rooms. When testing the task routes we did find two kiosk maps, recessed into the wall and presented in grey on a black background. One was located at the base of an escalator, and the other in the pedway leading from an adjoining hotel to the WTCC.



(a) (b)

Fig. 13.3. (a) Pedway (indoor walkway) system map. The arrows show the general flow of the experimental tasks. (b) The Halifax World Trade and Convention Centre (WTCC).

Design decisions:

- Rather than run the study using the two existing kiosk map locations, we decided to place a kiosk map by the elevators at each level of the convention centre. This included the main lobby.
- It was difficult to order the floors on the WTCC map as levels were not numbered. We decided to identify the three convention levels as “lower, main, and upper” on our kiosk map.

There is no kiosk map as you enter Scotia Square mall from WTCC via the pedway, although there are kiosks in the center of the mall and at the street entrances (Fig. 13.4). The kiosks in Scotia Square contain maps of both Scotia Square and Barrington Place Shops, which is across another pedway. There are no corresponding maps in Barrington Place Shops; only very small (8”x20”) maps located by the stairwells intended for service personnel, and shop directories without maps placed on the wall along two corridors. This is likely a design decision as Barrington Place houses primarily boutique and tourist shops.

Design decisions:

- In addition to the public kiosk map in the centre of Barrington place mall, we placed kiosk maps at the pedway entrance to Scotia Square mall from the WTCC and the pedway entrance to Barrington Place shops from Scotia Square.
- We decided to maintain the existing kiosk design, that is, to show both malls on our kiosk map.

(a) (b)

Fig. 13.4. (a) An entrance to Scotia Square Mall. (b) Outside Barrington Place Shops.

We found that defining steps in the routes was fairly straightforward. As has been shown to occur in studies examining how people specify routes (for example, Denis, Pazzaglia, Cornoldi and Bertolo 1999), we agreed most times when identifying critical landmarks. We included landmarks along a path to both identify turns/decision points and to reassure participants that they are going in the right direction in longer, uninterrupted paths. While display resolution was a factor in determining which landmarks to include, a key aim in the interface design for the purposes of our study was to reflect the presentation style of the kiosk maps for the same region. This had the greatest impact on how routes were presented. The mobile phone interfaces are described in section 13.5.1.

13.5 Experimental design

We conducted a controlled, scenario-based user study (an experimental simulation), utilizing interactive design mockups combining kiosk maps with personal mobile devices in a realistic setting. Under consideration was the impact of interactivity and presentation on acquired survey, landmark and route knowledge. This was accomplished by varying the level of interactivity between the mobile device and the kiosk maps, and varying the presentation provided on the mobile device itself (the various interface configurations are detailed in section 13.5.1). We hypothesized that:

a) direct interaction with public kiosk maps would promote survey and route knowledge, and
b) reflecting the presentation used in public kiosk maps in the mobile phone interface would reinforce the kiosk map image, and thereby increase the effect in (a).

We were also interested in observing how participants would integrate other aspects of the environment, such as signage and spatial layout, in their wayfinding strategies, beyond resolving their environment with the detail provided in the route description on the mobile device. While we couldn't control these elements, our experimental settings varied in the amount of signage, placement and purpose of maps, the presence of clear landmarks, and spatial layout. We captured experimenter observations regarding how these cues were used by participants during the navigation tasks. Finally, we captured participant ratings and comments regarding the effectiveness of the various technology configurations.

There were two experimental factors: the presentation format on the mobile phone, and the method of retrieving route information. Each factor had two levels. The phone presentation was either *textual* or *paged*, described below. The method of retrieving route information was either through interacting directly with a kiosk map (as described below), or by entering a query on the phone itself. This yielded four conditions, crossing the two presentation formats and the two interaction techniques. The study employed a balanced within-groups design, such that each participant group conducted a task in each of the four conditions. The condition orders were balanced by interaction technique, then presentation. Tasks were always presented in the same order, to maintain narrative flow. Each task began at the final destination of the task that preceded it. A within-subjects design permitted us to capture comparative evaluation from participants, and to observe how the different configurations impacted the navigation patterns of participant pairs. This proved to be important given the variety of navigation styles observed.

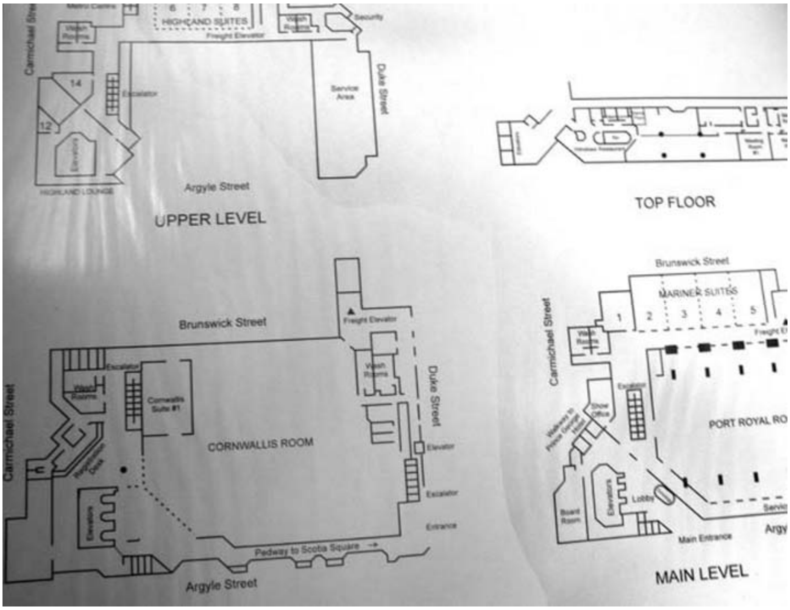


Fig. 13.5. Portion of the (3.2'X3.2') kiosk map for the WTCC. A similar map for Scotia Square/Barrington Place, and a smaller map showing the pedway system were also created.

13.5.1 Materials

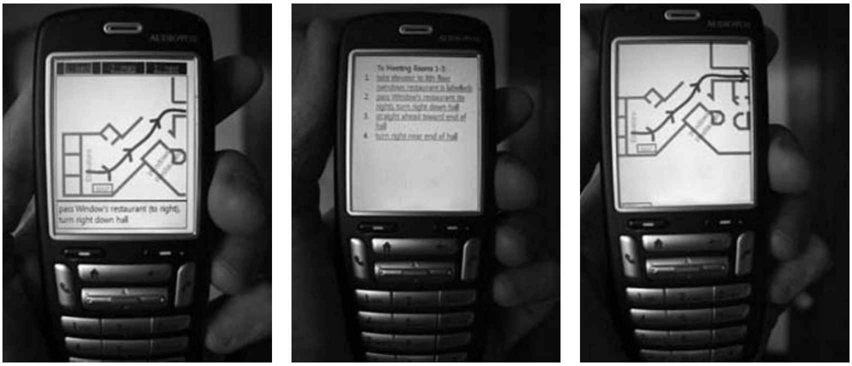
13.5.1.1 Kiosk maps

We created kiosk maps for the WTCC and Scotia Square/Barrington Place Shops (Fig. 13.5). Both maps were derived from the maps already present in the buildings, but employed a consistent line width and a two-colour presentation, to reduce differences in navigation success or participant ratings due solely to presentation. Names and markings closely followed those on the original maps: names for rooms and some landmarks are provided directly on the WTCC map, while a numbering scheme maps store locations to their names in an adjacent directory on the Scotia/Barrington map.

13.5.1.2 Mobil device interfaces

Participant pairs were given a single mobile phone (an Audiovox SMT5600) to share throughout all four tasks. Two interfaces were evaluated on the mobile device: the *paged* interface combines map segments and textual route descriptions on a single screen, and provides the ability to page through the steps along the route (see Fig. 13.6). The *textual* interface displays the entire textual route description as numbered items in a single page. Both interfaces provide access to a scrollable map outlining the route to take. The map image is scrollable in two dimensions using the phone's jog dial. From the textual interface this map view is accessed by first selecting a step in the textual description, which brings up the map view centered on the corresponding section. Similarly, pressing '2' in the paged interface brings up the corresponding section of the scrollable map.

The textual descriptions are identical in both the paged and textual interfaces, as are the maps in the paged and scroll views. The maps used are smaller versions of the kiosk maps, modified slightly as follows. A continuous blue line shows the route participants should take to their destination. Details not deemed useful for the specific route are removed from the electronic versions of the map, such as store numbers, while landmarks referenced in a route description are named directly on the electronic map if not already present on the kiosk map. Finally, labeled green squares identify locations along the route where kiosk maps are located.



(a) (b) (c)

Fig. 13.6. The mobile route application interfaces. (a) The *paged* interface, providing a map image and textual description at each phase in a route. (b) The *textual* interface, providing the route description as a numbered list. (c) The *scroll map*, traversed using the jog dial. This was accessible from both (a) and (b), opening to the corresponding location on the map.

13.5.1.3 Interactivity between the kiosk maps and the mobile device

In two conditions participants interacted directly with the kiosk map (termed the *pointing* conditions). Routes are requested by selecting a destination or series of waypoints in sequence on the kiosk map, using the mobile device as a pointer. The resulting route involves visiting each of the selected waypoints in the sequence that they were selected (we make no assumptions about intent and do not try to optimize the route). The result of the interaction was simulated using a "Wizard of Oz" approach. That is, rather than implement a truly interactive map, we instead mocked up the interaction by simply manually setting the corresponding route on the phone after the participants had made their selection using the kiosk map.

The interactions suggested by each task are basic, involving selecting one or possibly two waypoints at any given time. In addition, the 'Wizard' applied a liberal interpretation to the map interaction, such that in most cases the results were appropriate for the task. This was appropriate given the emphasis on the integration between the mobile interface and the kiosk maps *per se*, and not on the specific details of the interaction paradigm.

In the remaining conditions there was no direct interaction with the kiosk maps (the *non-pointing* conditions). In these conditions we made participants aware of the availability of the kiosk maps for use as static maps. Participants were told that route information was selected directly by using an application installed on the phone. Instead of doing this themselves, we had participants wait a small amount of time to reflect the time taken to query using the phone.

13.5.2 Tasks

During the experiment participants were told to imagine that they were conference attendees. This was intended to give some thematic consistency to the tasks conducted, and also to encourage participants to consider how they would approach the specific tasks in a real situation. We first walked through a training task with the participants, which involved going from the conference centre lobby to the registration desk.

The first two tasks were situated in the WTCC. While no conference was being held in the centre at the time, there were small groups of people throughout the complex. The first task involved meeting a friend outside a meeting room on the top floor of the WTCC, and going from there to a room on the main floor where the first session of the conference was being held. The second task required participants to go to their hotel (connected to the WTCC by pedway), and then return to a room on the upper level for the next conference session.

The remaining two tasks were situated in the two shopping malls. Trials were conducted during regular business hours, so the malls were quite busy. During task three, participants moved from the conference centre into Scotia Square mall. The conference was finished, and they wanted to visit travel agents to inquire about local tour packages. The final task continued the theme of shopping for tour packages. After having visited the two agents in Scotia Square in the previous task, participants were asked to go to a third travel agent located in the adjacent Barrington Place Shops.

13.5.3 Population

A total of 24 participants took part in the study, constituting 12 participant pairs. A requirement for participation was to be unfamiliar with the buildings used in the study, and participants were recruited in pairs. No other constraints on the population were made. Pairs were self-selected and included spouses, friends and colleagues. We conducted the study in pairs to encourage natural dialog during the tasks, which was recorded and used in analysis. While in most cases the general scenario (attending a conference) was reasonable for the specific pair, we did not require this to be so. Participants came from the university community primarily, with 8 female and 16 male participants, and ranged between 18-35 years of age. 6 of the 12 pairs were mixed gender.

13.5.4 Measurement

Participant pairs were accompanied by a facilitator and an observer during the trials. The facilitator recited task descriptions, answered questions when appropriate, and acted as 'Wizard' in the pointing conditions. The observer attempted to capture the style of communication between participants, the amount and nature of mobile phone use, the amount of kiosk map use, and how the environment was incorporated into the wayfinding tasks. While a coding sheet was used, these observations were necessarily qualitative in nature. Microphones were attached to participants in order to capture a more concrete record of their conversations during the tasks. The audio was used to clarify and support observational data.

In addition to audio and experimenter observation, participants completed a questionnaire at the end of each task, asking them to assess the effectiveness of various parts of the interface used during that task. After all tasks were completed participants answered a short summary questionnaire, asking for high-level evaluations of the interfaces they used.

Participants then completed a spatial knowledge questionnaire. The questionnaire was separated into two portions: the first done as a group, and the second done individually. Each question pertained to a single task. Three kinds of questions were in the spatial knowledge questionnaire: trace a route on one or more maps, identify locations highlighted on maps, and identify the locations on the maps corresponding to where a set of photos were taken. The questions types were designed to assess the level of acquired spatial route knowledge, landmark placement knowledge, and the level to which they attended to their surroundings, respectively. Questionnaire scores were collected per participant and per pair, for each task and question type. These provided a measure of the impact the experimental conditions had on acquired spatial knowledge.

13.6 Study results

13.6.1 Overall results

As illustrated in Fig. 13.7, participants responded positively to most aspects of the mobile phone interfaces in terms of satisfaction and perceived utility, across all tasks. For each task, the average ranking of all interface features was between 4 and 5 on a five-point Lickert scale, 5 being best. The one exception was the utility of the kiosk maps, for which many participants in the non-pointing conditions gave a 3 for neutral, having chosen not to use the kiosk maps at all in those conditions. Participants may have been influenced in their ratings by a novel experience, or a desire to respond positively to what they interpreted to be designs created by the facilitators.

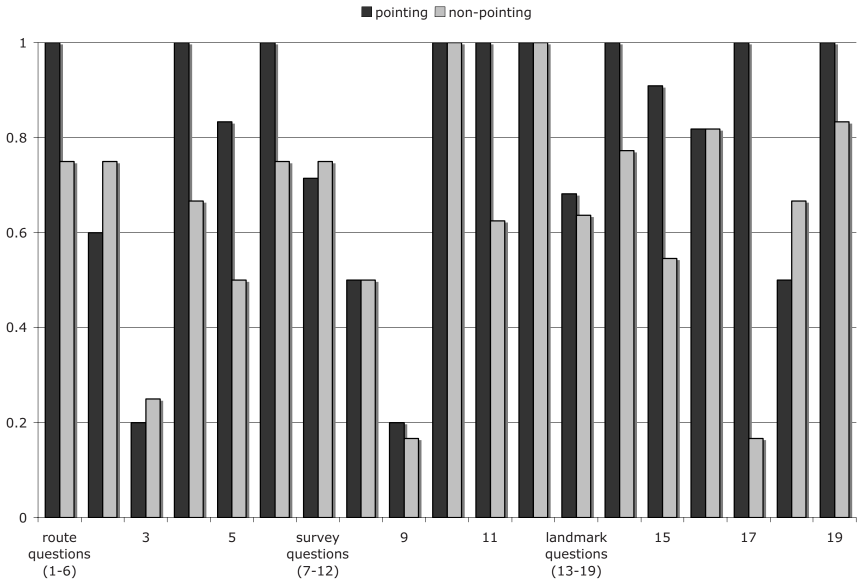


Fig. 13.7. Recall questionnaire results showing the proportion of correct responses to total responses for each question, for the pointing and non-pointing conditions. Participants in pointing conditions performed significantly better overall in questions involving tracing routes (U=26.5, p<.01).

Significant differences between interface/interaction conditions are discussed for each task below. One important general trend identified was that the pair recall score on questions testing route knowledge shows a significant main effect of interaction style (pointing vs. non-pointing) (U=26.5, p<.01), with a higher mean score for the pointing conditions.

Overall, task times were similar across conditions, but varied between tasks and participant pairs. Task times were lower on average for the textual interface across all tasks, however the difference was not significant (U=210, p>.1). All tasks were completed successfully by all participants, however, the time taken to complete tasks varied, with the slowest group taking up to 2.5 times as long as the fastest group on a given task.

13.6.2 Results by task

Because this was a naturalistic study in which we collected a range of observational data, each task is given a narrative characterization first in this section, followed by results obtained in the ranking measures and recall test scores that are peculiar to each task.

13.6.2.1 Task one: meet a friend then attend first conference session

For this task we observed the most instances of lostness (eleven, versus one or two cases for each of the subsequent tasks). Since the interface changed somewhat in subsequent tasks we don't interpret this as being due to learning the interface alone. Participant pairs may additionally have required time to establish working arrangements and general strategies for the wayfinding tasks. One specific decision point also had an impact, involving selecting a path from the main lobby of the convention centre. This caused some lostness for six of the twelve participant pairs.

A significant difference between pointing and non-pointing conditions was observed regarding the utility of external cues like signs and landmarks (mean 4.8 non-pointing versus 4.2 pointing, (U=39, p<.05)). A corresponding difference was not found when participants ranked the utility of the kiosk map, however.

13.6.2.2 Task two: stop at the hotel then attend another session

The second task was similar in certain respects to the first task: they both involved navigating around the main lobby area, and then into the conference floor. Some participants were able to recall the location of the pedway to the hotel from being lost during the first task, although not all participants remembered where they saw it. Participants appeared more comfortable navigating the conference floor in general, having gained exposure to the general layout in the first task. For example, participants 15 and 16 were able to hold an unrelated conversation while navigating during this task, whereas they showed careful concentration in the first task.

A significant difference was found in the perceived utility of the information on the phone between the textual (mean 4.8) and paged (mean 4.1) presentations (U=33, p<.05). This was especially so when the textual presentation was used alongside the kiosk map (pointing condition). In this condition all participants gave the highest rating (5) for the utility of the phone information, and the kiosk map was also rated highly (mean 4.67).

"The phone had simple directions. The Kiosk map was well laid out" – Participant #10

13.6.2.3 Task three: comparison shopping for tour packages

In the first phase of this task participants located the pedway from the WTCC to the mall. The pedway was not part of the mall or the convention centre. When in the pedway, bringing up the scroll map showed a high-level map of the pedway system, at a larger scale than the scroll maps in the WTCC (Fig. 13.8). While this was an accurate reference to the kiosk map relevant to the area, it was problematic because it broke the internal consistency of the route application on the device, especially for those who had not seen the kiosk map previously (i.e. most in the non-pointing conditions):

"the final map … was strange or a different scale and was difficult to figure out what it was. Once we did figure out what it was, it was of little use" – Participant #24

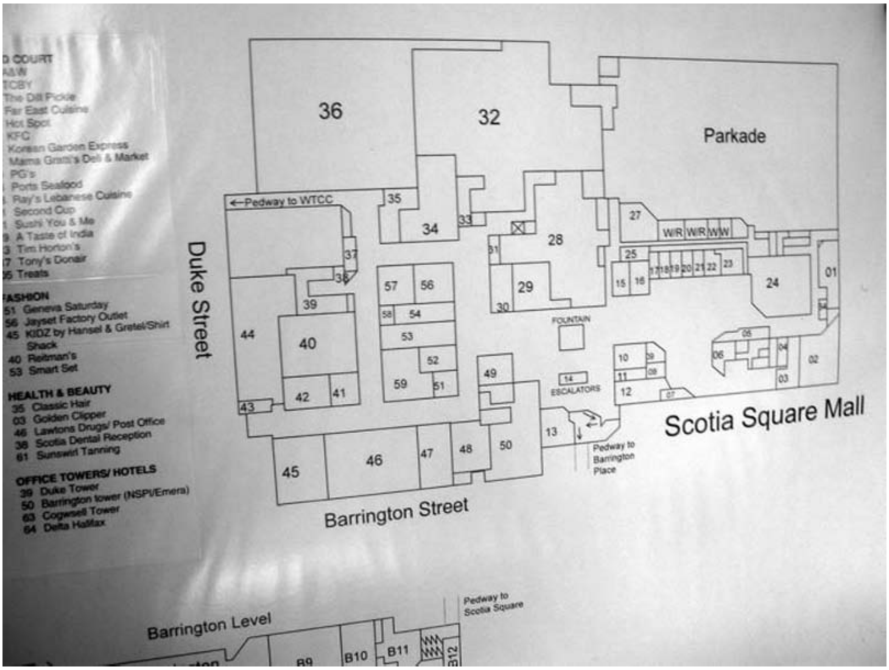


Fig. 13.8. Part of the kiosk map of Scotia Square and Barrington Place Shops. Stores are numbered on the maps, and linked to directories on the kiosk.

Participant pairs in the pointing conditions, having had exposure to the pedway kiosk map, fared better on the corresponding location placement question in the spatial knowledge questionnaire, which asked them to situate an image of the

WTCC-Scotia Square pedway (6/6 pointing vs. 1/6 non-pointing pairs answered correctly ($U=3$, $p<.01$)).

Once in the mall, participants could no longer use signage to help them navigate. We observed that participant pairs who were previously making use of signage shifted strategies and concentrated more fully on landmarks shown on the map view or route description. Participants were also able to effectively use aspects of the mall layout to navigate. Many participants visually scanned the open area of the mall and noticed the second travel agent from a distance. In pointing conditions many participants arrived at the first agent and realized there was no map there to query for the location of the second travel agent. Many participants in this situation instinctively walked toward the open area in search of a map, a wider view, or an information kiosk. This was despite the fact that they knew there was a map at the entrance to the mall from the WTCC pedway, which would require them to backtrack. Others scanned the scrollable map on their phone to locate the second agent. All participants in the pointing conditions gave the kiosk maps the highest rating (5) in terms of ease of use. This is despite the fact that most participants in these conditions were left without route information after the first travel agent because they selected only one on the kiosk.

13.6.2.4 Task four: last chance for a tour package

At the outset of this task, participants had to move from Scotia Square mall to Barrington Place shops by pedway (Fig. 13.9). This time the pedway was a short, above-ground walkway crossing a busy street, rather than a long meandering underground tunnel like the one connecting WTCC to Scotia Square. Many participant pairs verbally expressed uncertainty or exhibited lostness when locating the pedway, regardless of condition. This was even though it was located within 30 metres of the task's starting point. Once in Barrington Place participants generally did not have difficulty reaching their destination.

9 of 24 participants explicitly commented that the kiosk map did not provide much support for this task.

"The kiosk map was not as helpful for this task I think. We had to make quite a few turns so the phone was better" – Participant #2

Participants viewed the kiosk map at one of two locations: in Scotia Square before crossing the pedway, or in Barrington Place after crossing the pedway, both at the beginning phase of the task. Even though the route involved few difficult decision points, many participants expressed that they considered the route to be complex, as it involved going up and down levels, doubling back around escalators and along corridors with bends. When the phone interface was textual, fewer participants were able to place landmarks correctly on the map in the spatial knowledge questionnaire. A difference in score for the corresponding landmark placement question was found between the paged and textual interfaces (10/11 correct vs. 4.5/11 correct, ($U=28$, $p<.05$)).

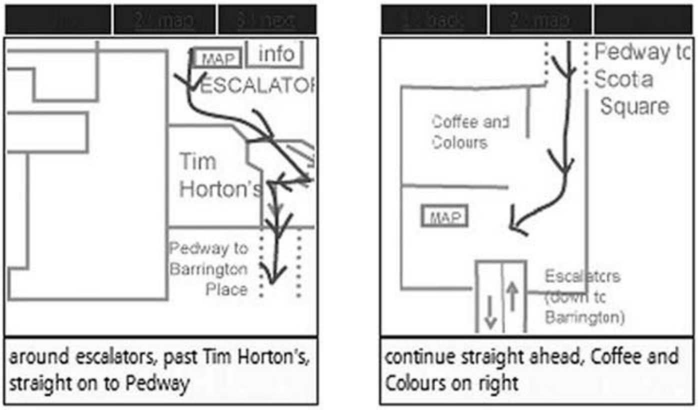


Fig. 13.9. The locations of the kiosk maps for task 4 shown here on the paged interface. The complexity of the remaining route made it difficult for those using the textual interface to relate back to the kiosk map.

13.7 Analysis and discussion

13.7.1 Designed elements

While we did observe a difference between the textual and paged interfaces in a landmark placement question pertaining to the fourth task, we did not observe a general trend across tasks, nor did we detect an interaction between mobile interface (textual or paged) and interaction technique (pointing or non-pointing). Therefore, we do not conclude that reflecting the kiosk map presentation on a mobile route application provided any benefit for spatial awareness. For more complex routes such as the one encountered in task four, graphical cues on the mobile device may serve to better relate the route to the map presentation. None of the participants specifically stated that they found the consistency between map views (on the kiosk maps and on the device) to be beneficial. Instead, one participant said they found it hard to relate the kiosk map with the map on the device, and several participants mentioned that the kiosk map was unnecessary, went unused, or wasn't detailed enough to be useful. By contrast, one pair felt strongly that the kiosk map was all that was needed, another pair also navigated using just the kiosk map, and others commented that the kiosk map was clearly organized. Given such a variety of perspectives it is hard to assess the impact from the user's perspective of integrating the kiosk map view and the phone map view.

There was a significant difference in spatial awareness test scores for route-related questions between the pointing and non-pointing conditions, regardless of the interface presentation used on the mobile phone (paged or textual). That is, interacting directly with the kiosk maps seems to have promoted a better spatial awareness of routes relative to the map. Because this was a naturalistic study, we did not require participants to look at the kiosk maps in all conditions. In fact, five participant pairs did not refer to the kiosk maps at all in the non-pointing conditions, since they were not required to do so in order to retrieve route information. As a result, we cannot conclude that interacting with the kiosk maps *per se* led to an increase in spatial knowledge acquisition. This may be due simply to exposure to the map itself. In the pointing conditions the actual time spent looking at the kiosk map was often quite short, and focused on finding the destination in large part. However, three of the twelve participant pairs also spent time trying to visualize a route to the destination using the kiosk map before pulling route information onto the phone in these conditions. Comments regarding the pointing interface were largely positive. In the non-pointing conditions, a kiosk map was visited in total nine times across all pairs throughout the study, with five pairs never looking at a kiosk map in these conditions. This suggests that providing route information directly on a phone can inhibit the use of kiosk maps, thereby potentially impacting the ability to relate a route to these maps. It should be noted, however, that while the direct interaction conditions did require participants to request routes using the kiosk maps, it did not increase the tendency to refer to these maps beyond what was necessary to retrieve route information. In the pointing conditions there were only three recorded instances of participants viewing a kiosk map for reasons other than requesting a route. When a task involved two or more phases, participants in the pointing conditions did not always retrieve route information for all phases from the kiosk maps. When one part of a task was complete and participants realized they would need to query a map again to get the next destination, several participants simply used the phone-based map, or surveyed the environment instead to help them find their destination, especially in Scotia Square where the last kiosk map used was a considerable distance away. Others found a kiosk map or remembered where one was, but rather than request a route simply memorized the location of the next destination relative to their current position.

The most common complaint about the kiosk map interface was that once at a destination they could not request a route from where they were standing. One possible solution to this is to permit the selection of several stops along a route. We have observed in prior studies that participants are quite adept at more complex queries after a small amount of training (Reilly et al. 2005). We had designed built routes on the mobile device to support this in the experiment, and in pilot testing participants had no trouble selecting multiple destinations in sequence to express a route with several stops after being shown how to do so. However we did not demonstrate how to do this in the study, and no participants seemed to consider that possibility when interacting with the kiosk maps.

13.7.2 Environmental elements

At least as important as the designed elements of our study to the navigation patterns observed were environmental elements such as signage, landmarks, spatial structure and dynamics. All of these contribute to the navigational infrastructure of a space, and any application designed to support navigation should consider these factors. We explicitly encouraged participants to make use of any cues in the environment when completing the tasks.

Signage was especially important in the WTCC setting. As described previously, signage was abundant in the convention centre floor, but virtually non-existent in the lobby areas of each level. Most pairs made some use of signage during the tasks set here, however the amount and style of use varied widely. Two groups relied on landmarks and signage exclusively for large portions of tasks in the WTCC, first looking at a map (either a kiosk map or on the phone) then using cues in the environment to navigate. Other groups relied on signage alone for small portions of a route, but the most common strategy was to use signage and other cues in the environment to reinforce or clarify information presented on the phone. In a few cases, the phone information helped to clarify signage, as when a sign was misinterpreted. At the other extreme, a couple of groups ignored signage for at least one task, focusing instead on the phone information. In areas in the WTCC where there was no signage, participants naturally switched their attention to landmarks and the phone interface. Without signage, the phone interface became more critical for navigation.

"phone interface particularly useful between floors e.g. --> how to get from meeting to mariner room" – Participant #6

Most participants were quite resourceful, adapting their strategies based on the environmental cues available. While the WTCC offered pervasive signage on the conference floor, the Scotia Square mall gave only typical mall signage, showing the way to washrooms, telephones, anchor stores and facilities attached to the mall such as hotels and office buildings. The majority of participants shifted to identifying landmarks referenced on the phone route display, while one pair simply memorized the route from the kiosk map at the mall entrance. When the final task brought participants into Barrington Place, a few participants shifted again to make some use of the spare, understated signage in the building, which included a store directory.

Landmarks were used by participants throughout the four tasks in this study. When landmarks were referenced in the phone route description, participants generally tried to identify the landmark in the real world, unless they had already established their location. For example, in the WTCC there is a set of escalators linking the main conference levels. In their direct path, participants generally acknowledged this landmark whether or not they were changing floors, while the adjacent Show Office, to one side of their direct path, was only acknowledged by three participants, even though all participants had a direct reference to this landmark in the route description. Landmarks also played an important role when lost, and when participants were looking for a kiosk map to interact with. In the WTCC the main lobby was used as a reference point when trying to determine orientation, especially at the start of a task. In Scotia Square the fountain was a recognizable point of reference when trying to locate a kiosk map or the information booth. Even when participants had no indication that there was a map in the area, the central, open space in which the fountain was located seemed an appropriate place to start looking.

The dynamics and structure of the buildings also played a role in wayfinding. When the door into the convention centre floor from the lobby was open, participants had less trouble determining the direction of their route as going through the doors: when the doors were closed there was considerably more uncertainty. The flow of people walking towards and from the pedway between Scotia Square and Barrington Place was another cue that helped some participants find the pedway. One pair of participants finally decided to go down an escalator in Barrington Place shops not because they were satisfied that they knew their route, but because a lot of other people were taking the escalator.

13.7.3 Integrating the environment in mobile map applications

Our main observation when designing the experiment was that the presence of kiosk maps (or any maps) in buildings is mixed. In the WTCC kiosk maps were practically hidden, while in Barrington Place they were non-existent. Maps of the pedway system exist on some stretches of the pedway but not others. Maps in Scotia Square were situated in street entrances, not pedway entrances. When assessing potential study locations on campus we had an equally varied experience. A designer cannot assume any level of infrastructure when designing mobile applications for navigation support. Designers may consider placing (as we did) additional kiosk maps to better support a route application; however, this may not be possible or might be at odds with the design decisions made for the public navigation support situated in a given environment.

In other cases, such as the campus buildings we assessed as candidate experimental settings, the existing support may be sufficient enough for the majority of navigation tasks. In such cases, designers might focus on supporting memory (e.g. recording room numbers from a directory for later retrieval), or atypical navigation tasks (e.g. locating a one-time event in a large library).

In our study there were three route decision points that were challenging to navigate for many participants. In all three cases there were several candidate paths, and signage was either not present or not obvious. In addition, two of the three points involved transitioning from one building to another. Mobile application designers should be aware of points at which existing navigational cues fail, and might focus on supporting decision making at these junctures.

Finally, we observed a variety of usage patterns when our participants conducted the experimental tasks. Some relied primarily on environmental cues, paying little or no attention to the mobile route application. Others used environmental cues like signage to corroborate the route information on the mobile device. Still others focused on the route information on the device primarily, looking to the environment only for those elements that were referred to in the route description. Individual and group differences in navigation approach impact how environmental cues are used, and ultimately how mobile applications can best support navigation tasks. For example, it may be more important to quickly retrieve route information about the current location if a user refers only periodically to the device, while a user who follows the route description closely would likely benefit from the ability to review aspects of the route.

13.8 Conclusion

The results obtained regarding the impact of integrating kiosk maps into mobile wayfinding applications are mixed. A significant effect was found for reflecting a kiosk map's presentation in a mobile route application for a landmark placement question pertaining to one of the four tasks in this study. This task involved a labyrinthine route, and many participants who used a graphical route display expressed having to consult it frequently in this task. This result does not show an interaction between mobile interface and the use of the kiosk maps, and is isolated to a single task. The result does not give strong support to the hypothesis that reflecting a kiosk map's presentation in a mobile route display enhances the ability to relate route details back to the kiosk map.

The study's results do provide stronger evidence that encouraging kiosk map use in a design can promote the acquisition of useful spatial knowledge relevant to routes taken. Specifically, when participants were required to interact with kiosk maps, they were able to transcribe routes in the spatial awareness test with greater accuracy on average, than when tasks were completed without interacting with the kiosk maps. We conclude that exposure to the kiosk map benefited our participants, however the choices made by participants in this experimental simulation do not permit us to distinguish between interacting directly with the kiosk map by pointing, and merely scanning the map visually.

We have presented results from a study examining how mobile wayfinding tools are used in conjunction with existing navigational infrastructure. As with any naturalistic study, some precision was sacrificed in order to gain realistic observations. In addition, the specific experimental context, including the buildings and tasks chosen, have had a considerable impact on our observations. However, this serves the point of this chapter, that is to say that environmental cues above those typically included in route descriptions on a mobile application can have a considerable impact on how such an application is used, and on what benefit a user will derive from it. We have observed that landmarks, signage, and kiosk maps are all important tools that were used by participants alongside or instead of the mobile application. The quality and consistency of existing public navigational aids varies widely. Designers must carefully consider the effectiveness of existing aids before designing mobile applications around them. Further, the navigational behaviour we observed was influenced not only by navigational cues in the environment, but also by group strategies, expectations, and the novelty of an environment. Being aware of existing navigational aids can only be one aspect of effective mobile wayfinding applications design.

Acknowledgements

We are greatly indebted to Jennifer Milne of Dalhousie University's GIS Centre for her work creating the maps used in this study. We also sincerely thank the WTCC and Halifax Developments, Inc. for permission to use their buildings. Finally, we thank the reviewers for their valuable comments on an earlier version of this manuscript. The chapter is much improved as a result. This research was supported by the National Science and Engineering Research Council of Canada (NSERC), Intel Research and Dalhousie University.

References

Agrawala, M., Stolte, C. (2001): Rendering Effective Route Maps: Improving Usability through Generalization. In *Proceeding of SIGGRAPH*, Los Angeles, California, August, 2001, pp. 241-250.

Bieber, G., Giersich, M. (2001): Personal Mobile Navigation Systems – Design Considerations and Experiences. *Computers and Graphics*, Vol. 25, pp. 563-570.

Borchers, J., Deussen, O., Knörzer, C. (1995): Getting it Across. *SIGCHI Bulletin*, Vol. 27 No. 4, pp. 68-74.

Butz, A., Baus, J., Krüger, A., Lahse, M. (2001): A Hybrid Indoor Navigation System. In *Proceedings of IUI*, Santa Fe, Mew Mexico, January 2001, pp. 25-32.

Christian, A.,D., Avery, B.L. (2000): Speak Out and Annoy Someone: Experiences with Intelligent Kiosks. In *Proceedings of CHI '2000*, The Hague, Amsterdam, April 2000, pp. 313-320.

Czerwinski, M., van Dantzich, M., Robertson, G., and Hoffman, H. (1999): The Contribution of Thumbnail Image, Mouse-over Text and Spatial Location Memory to Web Page Retrieval in 3D. In *Proceedings of INTERACT '99*, pp. 163-170.

Daniel, M. P., Denis, M. (2004): The Production of Route Directions: Investigating Conditions that Favour Conciseness in Spatial Discourse. *Appl. Cognit. Psychol.* Vol 18, pp. 57-75.

Denis, M., Pazzaglia, F., Cornoldi, C. and Bertolo, L. (1999): Spatial Discourse and Navigation: An Analysis of Route Directions in the City of Venice. *Appl. Cognit. Psychol.* Vol. 13, pp. 145-174.

Hommel, B., and Knuf, L. (2000): Action Related Determinants of Spatial Coding in Perception and Memory. In Spatial Cognition II, LNAI 1849, pp.387-398.

Huang, A., Pulli, K., Rudolph, L. (2005): Kimono: Kiosk-Mobile Phone Knowledge Sharing System. In *Proceedings of the 4th International Conference on Mobile and Ubiquitous Multimedia*, Christchurch, New Zealand, December 2005.

Johnston, M., Bangalore, S. (2004): Multimodal Applications from Mobile to Kiosk, W3C Multimodal Workshop. Sophia Antipolis, France.

Kaasten, S., Greenberg, S., and Edwards, C. (2002): How People Recognize Previously Seen Web Pages from Titles, URLs and Thumbnails. In *Proceedings of Human Computer Interaction 2002*: In J. Faulkner, F. Detinne (Eds) People and Computers Vol. XVI, BCS Conference Series: Springer Verlag, pp. 247-265.

Kray, C., Baus, J. (2001): A Survey of Mobile Guides. In Workshop on Mobile Guides at: *Mobile Human Computer Interaction '03*, Udine, Italy, September 2003.

Kray, C., Laakso, K., Elting, C., Coors, V. (2003): Presenting Route Instructions on Mobile Devices. In *Proceedings of IUI '03*, Miami, Florida, January 2003, pp. 117-124.

Kray, C., Kortuem, G., Krüger, A. (2005): Adaptive Navigation Support with Public Displays. In *Proceedings of IUI '05*, San Diego, California, January 2005, pp. 326-328.

Look, G., Kottahachchi, B., Laddaga, R., Shrobe, H. (2005): A Location Representation for Generating Descriptive Walking Directions. In *Proceedings of IUI '05*, San Diego, California, January 2005, pp. 122-129.

Lynch, K. (1960): *The Image of the City*. MIT Press, Cambridge Massachusetts, 1960.

MacEachren, A.M. (1995): *How Maps Work: Representation, Visualization, and Design.* The Guildford Press, New York, 1995.

Mackay, B., Watters, C., and Duffy, J. (2004): Web Page Transformation when Switching Devices. In *Proceedings of MobileHCI '04*, Glasgow, Scotland, September 2004: In S. Brewster and M. Dunlop (Eds.). Lecture Notes in Computer Science (Berlin: Springer), Vol. 3160, pp. 228-239.

May, A.J. Ross, T., Bayer, S.H., Tarkiainen, M.J. (2003): Pedestrian Navigation Aids: Information Requirements and Design Implications. *Pers Ubiquit Comput*, Vol. 7, pp. 331-338.

Meng, L. (2005): Egocentric Design of Map-Based Mobile Services. *The Cartographic Journal*, Vol. 42 No. 1, pp. 5-13.

Nivala, A., Sarjakoski, L.T. (2003): Need for Context-Aware Topographic Maps in Mobile Devices. In: Virrantaus, K. and H. Tveite (eds.), In *Proceedings of the 9th Scandinavian Research Conference on Geographical Information Science*, June 2003, Espoo, Finland, pp.15-29.

O'Neill, M. (1991): Effects of Signage and Floor Plan Configuration on Wayfinding Accuracy. Environment and Behavior, Vol. 23 No. 5, pp. 553-573.

Oulasvirta, A., Nivala, A., Tikka, V., Liikkanen, L., Nurminen, A. (2005): Understanding Users' Strategies with Mobile Maps. In *Mobile Maps 2005 Workshop on Interactivity and Usability of Map-based Mobile Services, Conference MobileHCI '05*, Salzburg, Austria, September 2005.

Peterson, M.P. (2004): Pervasive and Ubiquitous Public Map Displays. In *Proceedings of the International Joint Workshop on Ubiquitous, Pervasive and Internet Mapping*, Tokyo, Japan, September 2004.

Phillips, R.J. and Noyes, L. (1982): An investigation of visual clutter in the topographic base of a geological map. *Cartographic Journal*, Vol. 19 No. 2, pp. 122-132.

Reichenbacher, T. (2003): Adaptive Methods for Mobile Cartography. In *Proceedings of the 21st International Cartographic Conference (ICC)*, Durban, South Africa, August 2003, pp. 1311-1322.

Reilly, D., Rodgers, M., Argue, R., Nunes, M., and Inkpen, K. (2006): Marked-up Maps: Combining Paper Maps and Electronic Information Resources. *Pers Ubiquit Comput*, Vol. 10 No. 4, pp. 215-226.

Reilly, D., Dearman, D., Welsman-Dinelle, M., Inkpen, K. M. (2005): Evaluating Early Prototypes in Context: Trade-offs, Challenges, and Successes. *IEEE Pervasive Computing* (October-December 2005) Vol. 4 No. 4, pp. 10-18.

Sefelin, R., Bechinie,M., Müller, R., Seibert-Giller, V., Messner, P., Tscheligi, M. (2005): Landmarks: Yes; but Which? Five Methods to Select Optimal Landmarks for a Landmark- and Speech-based Guiding System. In *Proceedings of MobileHCI '05*, Salzburg, Austria, September 2005, pp. 287-290.

Tamminen, S., Oulasvirta, A., Toiskallio, K., Kankainen, A. (2004): Understanding Mobile Contexts. *Pers Ubiquit Comput*, Vol. 8, pp. 135-143.

Tom, A. and Denis, M. (2003): Referring to Landmark or Street Information in Route Directions: What Difference Does It Make? In *Proceedings of COSIT 2003*, LNCS 2825, pp. 362-374.

Weisman, J. (1981): Evaluating Architectural Legibility: Wayfinding in the Built Environment. *Environment and Behavior*, Vol. 13 No. 2, pp. 189-203.

Zipf, A. (2002): User-Adaptive Maps for Location-Based Services (LBS) for Tourism. In: K. Woeber, A. Frew, M. Hitz (eds.), *Proceedings of the 9th International Conference for Information and Communication Technologies in Tourism, ENTER 2002*. Innsbruck, Austria. Springer Computer Science. Heidelberg, Berlin.

14 Geographical Data in Mobile Applications Uses beyond Map Making

Ashweeni BEEHAREE, Anthony STEED

Department of Computer Science, University College London

Abstract. Mobile applications typically only exploit geographic data for the purposes of rendering of local area maps. These maps are an essential part of guiding applications, and a lot of work has been done on methods for rendering of clear, useful maps. However, with the rapidly increasing power of the mobile devices themselves and the increasing ubiquity of GPS positioning, much more can be made of the geographic data itself. For example, location-aware or location-based applications are becoming more common. In this chapter we present three main uses of geographic data. The first is using such data to support the description of regions on map that correspond to places in the real world to which location-based information might be attached. The second usage is for de-cluttering map data to ease the load on mobile applications as well as to improve usability. This is done by exploiting forms of visibility computation that can be done with 2D map data. The third usage is to again exploit visibility analysis to support the insertion and retrieval of geo-located data, using its likely region of use. Finally, as 3D geographic data is becoming more widely available, we briefly discuss the potential role for 3D map data in mobile applications. All these advances have raised new challenges for storage, retrieval and presentation of map data.

14.1 Introduction

In mobile applications, geographic data has usually been used solely to generate maps. This is not surprising since the majority of mobile applications have simply provided a new way to visualise geospatial data in a manner that is analogous to their paper-based counterparts. However, as navigational applications have become increasingly sophisticated, several other applications, such as location-aware applications and mobile games have emerged and increasingly there is a desire to support user authoring and annotation of maps within the applications. Mobile map-based application development has gathered a lot of momentum over the last decade, in various forms, ranging from navigation tools such as “TomTom” (TomTom, 2006) to games of the likes of “Can You See Me Now?” (Benford et al., 2006) and “Yoshi” (Bell et al., 2006).

The terminology around map-data and location-aware applications is slightly confusing. In the remainder of this chapter, when we refer to position we refer to a

2D value, usually reported by some tracking technology that refers to some coordinate system. Common coordinate systems include WGS84 (NGA, 2007) (a latitude and longitude reported by GPS units), and, in the United Kingdom, OSGB (Ordinance-Survey, 2007), a coordinate system of metres measuring north and east that is used in maps in the UK. Users will thus receive a reported position, but note that tracking technologies are inaccurate, so this is necessarily an approximation to where they actually are. The term location is sometimes used interchangeably with position, but we will use this term to refer to a static description of where an object, such as a building is. This might be as simple as a 2D value in some coordinate system, in which case we will refer to it as a location coordinate but equally it could refer to an area such as a building outline or post-code district that is defined by a series of 2D values that give its outline, in which case we will refer to it as a location region or simply region if the meaning is clear. In other literature, location can also refer to a symbolic location, which are names or identifiers that are uniquely identifiable and (relatively) static. This might be place names or something like WiFi network identifiers or RFID beacons. It is eminently possible to build a location-aware application that never refers to any coordinates in a coordinate system, but only responds when such symbolic locations are identified, for example the PlaceLab system (LaMarca et al., 2005). Of course such applications would not be able to present maps, without there being an effort to tie symbolic locations to some coordinate system, by, for example, associating every symbolic location with a location coordinate or location region. More in depth discussions of the role of position and location can be found in (Hightower and Borriella, 2001, Steed et al., 2004).

The design and generation of maps for small form-factor devices is complex because the screen size is small, and the map must be very clear to be readable under daylight conditions that the mobile device will be used. For an application for use in a specific area such as a museum or local tour guide, application developers might use a hand-crafted map (e.g. in “Can You See Me Now?” (Benford et al., 2006)). Such a labour intensive process would not work for general areas, or for generation of maps for more open geo-spatial applications where users can collaboratively edit or annotate maps (SOMA, 2006).

The second usage is for de-cluttering map data to ease the load on mobile applications as well as to improve usability. This is done by exploiting forms of visibility computation that can be done with 2D map data. The third usage is to again exploit visibility analysis to support the insertion and retrieval of geo-located data, using its likely region of use. The specific example we demonstrate is storing geo-located photographs by attaching them to buildings in the photograph, not just a location coordinate from where they were taken. Finally we discuss about the new possibilities for mobile maps that arise from the increasingly availability of 3D geographic data sets.

14.2 Authoring

Increasingly mobile applications go beyond presenting simple maps, to giving location-aware and context-sensitive information to the user. The presentation of information within a mobile application can be triggered by a wide range of predicates such as user interaction or timers. However once a tracking technology (e.g. GPS) is available, the application can be triggered once the device is near or within a pre-defined location. Defining these triggers is part of the creative task of authoring the application. There are many ways of authoring regions that can act as triggers. One example pair of authoring and client interfaces is the Mobile Bristol Toolkit (Hull et al., 2004). The authoring tool allows the author to load a map, and describe polygonal and circular regions over this map. These regions can have multimedia clips attached to them. Once saved and loaded into the client program, the multimedia clips are played once an attached GPS unit indicates the user is within the associated region. A different example is the authoring example that underpins the "Can You See Me Now?"-game (Flintham, 2005). In this system the author paints colours on to a raster map that overlays the area of the application. Each colour triggers or controls a different aspect of the application.

Another problem which emerged through experimentation (Beeharee and Steed, 2006a) is related to the consistency of map data with the real world. One can appreciate that it is a colossal task to maintain a map database for constantly changing cities such as London. This makes it difficult for authoring mobile experiences which depend on the accuracy of the map data.

This section will first describe a way to exploit map data in order to assist authoring of mobile experiences, and then go on to suggest how map data can be made accurate by tapping into user knowledge.

14.2.1 Location region marking tool

As discussed above, one of the most important facilities in mobile applications is the ability to trigger events based on location. There are several ways this might happen, firstly, the proximity to a physical real life entity, such as a building or a statue, can trigger an event. For example, in the Urban Tapestries project (SOMA, 2006), the user carries a PDA which is tracked by a GPS unit. There is an online database which stores annotations at location coordinates on the map. As they move around, a local map is drawn and the client software fetches annotations of nearby location coordinates and shows on the map as links that can be clicked on. In this case, locations are simple 2D coordinates, and "nearby", simply means close in distance. An alternative, as demonstrated in the George Square system (Brown et al., 2005), is to mark out location regions. In that application, the user is tracked by GPS, but this time they carry a tablet PC. As they move around they are shown web pages and links to media which are triggered when their GPS-reported position enters the location region.

Drawing individual regions for each annotation or event that one wants is time-consuming. What is of common interest to the authors of a mobile application is an extended footprint area around a particular entity such as a building. This is because when a user is approaching an entity the triggering of an event may be required before the user is actually within the footprint of the entity. For instance, in a pedestrian navigation application, instructions may be more relevant when approaching a building rather than on reaching it.

Secondly, it may be required to refer to a group of buildings or physical entities rather than individual ones. A common practice is to mark such regions manually, when a better job can be done by combining polygons from map data representing footprints of buildings, roads, etc into an area.

Thirdly, only part of a building may be relevant to an application. For instance, the part of a building that has an entrance is far more important than its back from the point of view of pedestrian navigation. Application developers can split existing footprints to denote such areas.

Note that in each of these examples, there is already geographic data available that can assist with the description of the location region: the existing building outlines that form the map can be exploited to speed up the authoring process. In (Beeharee and Steed, 2005), a tool was developed to assist developers to author mobile applications. The tool allows for both arbitrary description of polygons for the boundaries of regions as well as definitions of regions based on pre-existing polygons defining footprints of physical entities in map data. This is unlike automatic extraction of landmarks from map data (Elias, 2003), as it allows more flexibility in describing, for example, location regions in open spaces, or location regions that applied only to subparts of large buildings, see Fig. 14.1(a).

Using this tool, regions can be marked by drawing new polygons on the screen or by selecting building polygons from existing map databases. In principle, any vector information from the map database could be used as a basis for the description of a region boundary (e.g. road and pavement edges), but in the current implementation only whole buildings can be selected, or completely new polygons drawn. However the complete map data is drawn as an underlay to assist the user when marking of areas of interest and boundaries of location regions. This map data is the same dataset that is used to generate dynamic raster maps for this application (Fig. 14.2(a)). Based on the reported position of the user, the building polygons within a certain radius are automatically retrieved from the map database and displayed as candidates for marking as shown in Fig. 14.2(b) and 14.2(c). Whenever the user changes position, a new set of building polygons is potentially retrieved and displayed. This speeds up the marking process greatly. The radius of the area around the user for which building polygons are to be retrieved is specified by the user.

(a) Example location regions. Foci boundaries are indicated darker than other building contours. Icons indicate the type of resource, in this case building descriptions. These are automatically placed in the gravity centre

(b) Creating a new region based on the building contours. The user can then go back to the foci and test to find if boundary is visible from a user's viewpoint.

Fig.14.1. Editing foci and resource allocation

(a) The interface

(b) Building highlighted depending on user position

(c) Building polygons fetched on the fly from map database

Fig. 14.2. Map assisted marking

14.3 Visibility

Once we have a rich geometric data set containing location regions, and underlying map data, one new potentiality is the computation of visibility of locations from a given position, or from one location to another. In computer graphics, it is common practice to compute visibility of objects from the user's perspective using

geometric information of a scene. Map data can also be used to this end, and lead to many interesting applications that will be described in the next section.
(Cohen-Or et al., 2003) present an overview of visibility algorithms from a computer graphics research point of view. Although analytic solutions for visibility exist, because our system uses inaccurate positioning and editing processes, a probability-based approach is more applicable. Furthermore it may be required to know how much can the user see from his/her current position, thus requiring an estimate of how much of the user's view a building covers, and most simple analytic solutions give only a binary visible or not visible result. These issues and the limited computation capability of mobile devices suggest that simpler, sampling-based approaches are needed.

14.3.1 Visibility from a position

We have used a raycasting algorithm to compute the visibility of location region boundaries from a user's view volume (Fig. 14.3). This algorithm has the advantage of being easy to implement, and the one based on region visibility is a simple variant.

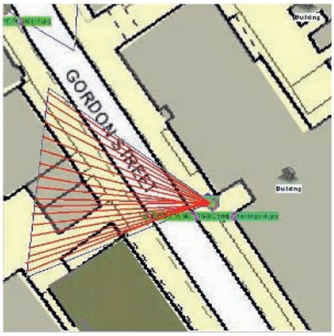


Fig. 14.3. Example of raycasting approach for associating location region to user viewpoint. The lines originating from the photograph icon (presenting the user viewpoint) indicate some example rays that are cast from the user's position.

The visibilities are calculated using the following steps. Firstly, a set F is populated with all potentially occluding map features (such as buildings) and location regions which intersect with the view volume of the user. About one hundred rays, with the user's position as their initial point, are created according to the expanse of the view volume. These rays are cast for each of the polygons in the set F.

If a location region is hit, a check is performed to ensure that there is no occluding map feature between it and the user's viewpoint along the same ray. This is done by comparing the distances of the intersection points of the occluding map feature and location region from the view volume's origin. If the region under inspection is the nearest one to be intersected by the ray, it's counted as a hit.

Otherwise it moves on to the next ray. The user's view can then be associated with all the regions that were hit.

The algorithm proposed here does not take into account heights of buildings, as these are typically not available in most map data sets. However, navigation applications on mobile devices are increasingly using 2.5D and 3D visualisation and we discuss some of the new potentials such data sets provide in section 14.6. As more accurate height data become readily available, the existing algorithm can be extended to take into account heights. The problem in 3D map space then becomes increasingly similar to the occlusion problems that are encountered in 3D computer graphics. In this case however, a statistical approach in computing visibility would be less demanding in terms of computational power, as compared to other 3D occlusion solutions.

14.3.2 From-region visibility

Because the user's position is known only inaccurately, for some purposes a from-region visibility algorithm is useful (Cohen-Or et al., 2003). We assume that some form of probability distribution is given for the user's position. Once again, a sampling-based method is appropriate because it can estimate likelihood of visibility no matter the shape of the position probability distribution.

The algorithm used to compute the visibility of the recommendations is as follows. A set P is populated with all polygons to be considered in the visibility computation. A view polygon around the view point is considered. This view polygon represents the uncertainty in the GS positioning. The view polygon can be chosen so that the user is within a certain probability of being within the view polygon. Obviously, the worse the uncertainty in the tracking system, the larger the view polygon will be, and thus the more likely that a target is visible.

Another set of polygons, R, is created corresponding to the recommended features. R need not be, but usually is a subset of P. The following steps are repeated for all polygons in R, referred to as target polygon.

1. From the set P, a new subset is defined as the polygons which lie within a rectangular region subtended by the viewpoint and the centroid of the target polygon. This subset of polygons is the set of potentially relevant polygons.
2. Select n number of random points within the view polygon and the target polygon. Note that the random points need not be sampled uniformly, but can be chosen to fit a probability density function that represents the tracking uncertainty.
3. For each ray from a point in the view polygon to the target polygon, find intersections with the set of relevant polygons. Search for intersection (for a line) is halted as soon as it intersects ANY polygon. The visibility confidence of a given target polygon is given by 1-(number of intersection found/number of rays considered).

This process is illustrated in Fig. 14.4, where five rays are shot, but only three hit the building, giving a confidence of 60%. However for the actual system 10 rays were used to determine visibility for each location region. The set P can be populated in two ways: using some set of region polygons found in an existing database of region polygons or by using the set of closed polygons from the Ordnance Survey map database.

Fig. 14.4. Visibility computation for the filter -A random point is taken in the region around the reported position (the circle). Another random point is taken on the region (a building outline in this case), and ray is defined between them. The thicker rays hit the target while the thinner rays are obstructed by buildings in their path.

14.4 Filtering and highlighting

So now that we can calculate what is visible to a user from their position, it is possible firstly to selectively present information to the user, and secondly, to inform the user to what extent certain features on the map would be visible in the real world from a given position.

14.4.1 Visibility Filter

In a pedestrian navigation application, it makes sense to present instructions based on features which are visible in the real world (Beeharee and Steed, 2006b). Similarly, in a tourist navigation system recommendations about places to visit can be filtered based on visibility (Beeharee and Steed, 2005).

A visibility filter was developed to work with the original George Square system (Brown et al., 2005). In this application, recommendations were presented to the users. The recommendations would have an icon and a location region associated

with each. The filter computed if a location region -which represents a feature of interest -is visible from the user's position and, if so, to what extent.
The algorithm presented in section 14.3.2 was implemented and used to filter out recommendations that were pushed onto the user's application based on their visibility from the user's viewpoint. A very desirable side effect of this process is the de-cluttering of the user viewing space -which is very limited on mobile devices.

(a) Unfiltered recommendation (b) Filtered recommendation

Fig. 14.5. Maps showing recommendations of places to visit. Building Icons denote physical buildings, while photograph icon represent the position from where photos of the buildings were taken.

Fig. 14.5 shows a simple example of filtering with Fig. 14.5(a) showing unfiltered recommendations and Fig. 14.5(b) filtered recommendations. Note the building at the top of the figure has been removed.

14.4.2 Highlighting recommendations at run-time

Another application of the visibility computation is to present context-sensitive information to the user. So for instance, a user may be interested in knowing how much of a certain building on the map would be visible from a position.
In the test application that was developed, it is possible for the user to extract additional information about the presented recommendations. For instance, upon selection of the building, which is thus highlighted Fig. 14.6, a tool-tip appears saying from which photographs the building can be seen and how much space in the photograph it occupies. This information can be particularly useful when navigating using landmarks or photographs (Beeharee and Steed, 2006b).

14.5 Photo-keying

Fig. 14.6. Highlight recommendation to the user

With the widespread availability of cameras, including ones on mobile phones over the last few years, there has been an explosion in number of photographs that people take. Organising such photographs has become a complex task, as chronological order is no longer sufficient to provide enough contextual information. Therefore, location information has become increasingly important.

We have therefore seen the emergence of photo-keying, in other words, the storage of photographs associated with map data based on the location at which they were taken. This section presents an approach which uses map data to refine photo-keying. Web sites such as Pixagogo (Google, 2007) allow users to create *Phot*o *Map*s by uploading photographs, along with a description and information about the location at which they were taken.

The location information can be used in a number of ways. Firstly, the location information would assist in storing the photograph along with the likes of geometric data in the map database. Secondly, it supports intelligent software systems to automatically select most relevant photographs from a library of geo-located photographs. Thirdly, it allows for innovative ways to visualise photographs on 2D and 3D maps.

The integration of geo-located photographs with traditional map data has recently generated innovative application such as NavPix (NavMan, 2007) and photo-based pedestrian navigation systems (Beeharee and Steed, 2006b). In NavPix, photographs are presented along with navigation instruction on a 2.5D map to motorists at specific locations to assist in the navigation task. With the widespread use of GPS device and the integration of tracking technologies in cameras and mobile phones, it is only a matter of time before most digital photographs will be geo-located automatically.

A further refinement is possible if orientation information about the camera is available. Combined with various camera characteristics, the view volume of the camera can be determined. Such information allows for the computation of visibility of buildings and features within the photograph itself using position visibility algorithm discussed in section 14.3.1. The visibility information can then be used to better associate photographs with existing information in map databases, rather than simply relying on geo-location. For example, if a photograph of a landmark, such as the Eiffel tower is taken from a distance, it would still be better to associate the photograph with the Eiffel tower rather than with the position at which the photograph was taken.

Orientation information can be inferred if the camera is tracked and its position is known over time. Recently, camera addons have become available that allow the capture of orientation of a camera. Exemplar systems in research have been implemented to demonstrate that it is possible to use the view volume information to select a photograph that best represents a landmark to a pedestrian on a mobile device in a pedestrian navigation system (Beeharee and Steed, 2006b) or to associate the photograph with the most relevant building in the map database (Beeharee and Steed, 2005).

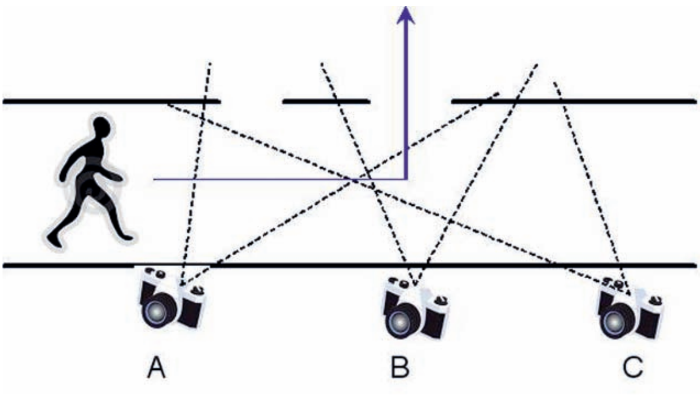


Fig. 14.7. Augmenting with routing information - the challenge

Consider figure 14.7, where the pedestrian is moving towards the point where a left turn is required along the route. Three camera positions (A, B, C) are considered, representing the actual photos shown in Fig. 14.8 from left to right respectively. The dotted lines in figure 14.7 represent the field of view from each camera position. One of the photos will be presented to the pedestrian along with navigation instructions. If the decision to choose a photograph simply relies on the location from which it is taken and its orientation, then any of the photographs would do.

However, by using existing map data and the visibility algorithm in section 14.3.2, it is possible to work out which photograph best represents recognisable landmarks as well as the percentage of the photograph occupied by a relevant geographical feature (in this example it is the gate) computed using 14.3.1 from the camera viewpoint. For instance, in this situation, the algorithm would recommend the photo taken from camera position B.

(a) Photo from camera position A

(b) Photo from camera position B

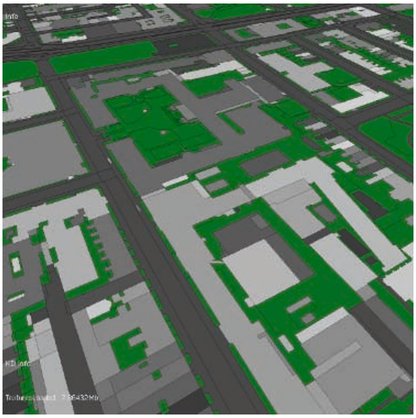
(c) Photo from camera position C

Fig. 14.8. Photos taken from three different camera positions

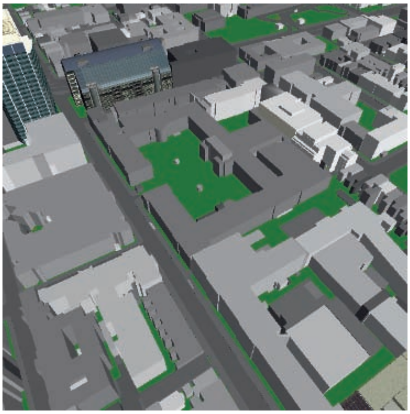
14.6 3D mapping

Map data has usually been supplied in 2D vector graphics format, as this is a good compact representation. Recently though, there has been a growing interest in 2.5D and 3D models, to supplement 2D data. Of course, there has always been a need for some aspects of 3D measurement within geographic applications, to retrieve feature heights, but now full 3D models are being captured and displayed: the popularity of applications such as Google Earth, and games that include models of well-known cities suggest that these data sets will become more readily available over time.

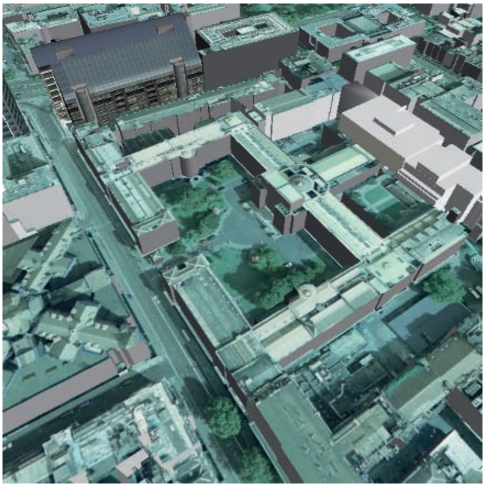
At the time of writing 3D data sets are either sparsely available or still proprietary in nature. There are some notable exceptions such as maps of building in Japan and major American cities, which are at least available to browse on application such as Google Earth. There are several approaches to making such models that depend on the types of information available (Hu et al., 2003). However, a common way is to use a fusion of 2D map vector map information, building heights taken from LIDAR (Light Detection and Ranging, essentially a laser-scan from an aircraft), aerial photograph and other geographic information about building types. The model is mainly comprised of simple extrusions of building footprints, but it does include correct roof shapes where these are easy to detect. LIDAR information comes at various levels of detail, but for the model discussed below it averages a sample every 1m in north and east. Similarly, as noted previously, the 2D vector map information contains some inaccuracies and ambiguities due to incorrectly modelled edges, or data which has not yet been revised. Some mismatches can thus be found between the aerial photography and the edges of features in the model.



(a) Visualisation of raw 2D vector



(b) Extrusion of 2D vector model to map data for a part of Central Lon-3D model using LIDAR heights and don incorporation of CAD models for two landmark building



(c) The same 3D model with aerial photography draped over

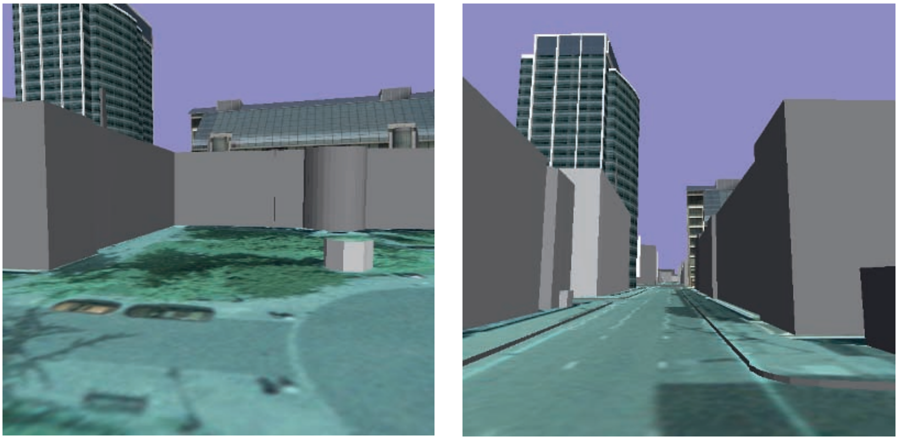
Fig. 14.9. Visualisations of a model of Greater London area

Fig. 14.9 shows visualisations of some the stages involved in the making of such a model for the Greater London area (specifically everywhere within the M25 motorway), generated by one of the authors. Note that the models are all of the same area of London for which the mobile maps drawn previously, in particular notice the ”lobster-shaped” building (actually the front quadrangle of UCL) in Fig. 14.1, 14.5 and 14.6, which is clearly visible in 2D and 3D in Fig. 14.9. Fig. 14.9(a), shows a perspective rendering of the original planar 2D vector map, with each building polygon outline given a unique shade of grey, roads and pavements in a uniform mid-grey, and other areas in green. Fig. 14.9(b), then shows a simple extrusion of the data to make prismatic models for the majority of buildings, but with two buildings which are obviously full models with facade detail and complex roof detail. Fig. 14.9(c) shows the buildings draped with aerial photography.

Techniques exist to capture facade imagery and reconstruct facade detail: this can be done with aerial photography if the facades are easily visible from the air, but considerable research is being done on autonomous research for capture from street-level (Fruh and Zakhor, 2003). However for the purposes of use for the mobile user, the simpler 3D model is very useful. For example, we discussed in section 14.3 how to do visibility sampling using 2D maps, and we noted in section 14.3.1 that the solutions didn’t encompass the full visibility solution. Using a 3D or approximate 2.5D model (i.e. one that only included height, not roof shape), we can rectify these problems. We can also visualise very quickly what buildings we currently fail to pick up. Fig. 14.10 shows two situations where the two landmark buildings from Fig. 14.9 are very highly visible in the frame, but where they would marked as invisible by a 2D visibility algorithm. In both, the user is close to ground level. In Fig. 14.10(a), they are inside the UCL quad, looking roughly north (up in Fig. 14.9(b)). The two landmark buildings are visible over the UCL quadrangle building itself because they are much higher. In Fig. 14.10(b), which is a rendering along a street, the two buildings are also visible in 3D, but in this case, their 2D visibility (i.e. the small fraction of the ground floor that can be seen) is so small, that they would be unlikely to be recommended or highlighted.

14.7 Conclusion

This chapter presented a discussion of five uses of mobile map data other than for generating maps. The need for more versatile exploitation of such data has been driven by the increasingly sophisticated mobile applications. Firstly, mobile maps can be used to empower mobile experience developers with tools that, for example, assist in intelligently marking of regions of interest by using map data. Secondly, much like the use of object geometry in computer graphics, mobile map data can be used in visibility calculations. Two techniques are presented for computation of visibility from a position and from a region. Thirdly, de-cluttering of mobile displays can be achieved



(a) View of landmarks from quadrangle (b) View of landmarks from the street

Fig. 14.10. Situations that affect visibility of landmark buildings

by filtering information using visibility and relevance inferred by map data. This improves both system performance and usability. The fourth usage addresses the issue with managing novel geo-located data generated by mobile devices. Though the section concentrates on photographs, the concept can be extended to other types of media as well as sensor data. Lastly, this chapter discussed some of the potential uses that the increasingly common 3D models of cities might have for mobile applications.

Acknowledgements

This work was supported by the UK projects, EQUATOR Interdisciplinary Research Collaboration (EPSRC Grant GR/N15986/01) and Advanced Grid Interfaces for Environmental e-science in the Lab and in the Field (EPSRC Grant GR/R81985/01). The vector data used was supplied by the UK Ordnance Survey.

References

Beeharee, A. K. and Steed, A. (2005): Filtering location-based information using visibility. In *LoCA*, pages 306–315. (Beeharee and Steed, 2006a) Beeharee, A. K. and Steed, A. (2006a). Exploiting real world knowledge in ubiquitous applications. *Personal and Ubiquitous Computing*.

Beeharee, A. K. and Steed, A. (2006b): A natural wayfinding -exploiting photos in pedestrian navigation systems. In *MobileHCI '06: Proceedings of the 8th conference on Human-computer interaction with mobile devices and services*, pages 81–88, New York, NY, USA. ACM Press.

Bell, M., Chalmers, M., Barkhuus, L., Hall, M., Sherwood, S., Tennent, P., Brown, B., Rowland, D., and Benford, S. (2006): Interweaving mobile games with everyday life. In *CHI '06: Proceedings of the SIGCHI conference on Human Factors in computing systems*, pages 417–426, New York, NY, USA. ACM Press.

Benford, S., Crabtree, A., Flintham, M., Drozd, A., Anastasi, R., Paxton, M., Tandavanitj, N., Adams, M., and Row-Farr, J. (2006): Can you see me now? *ACM Trans. Comput.-Hum. Interact.*, 13(1):100–133.

Brown, B., Chalmers, M., Bell, M., MacColl, I., Hall, M., and Rudman, P. (2005): Sharing the square: collaborative visiting in the city streets. *ALT.CHI 2005*, page under submission.

Cohen-Or, D., Chrysanthou, Y., Silva, C., and Durant, F. (2003): A survey of visibility for walk-through applications. *IEEE Transactions on Visualization and Computer Graphics*, 9(3):412–431.

Elias, B. (2003):. *Spatial Information Theory: Foundations of Geographic Information Science*, volume 2825, chapter Extracting Landmarks with Data Mining Methods, pages 398–412. Springer, Berlin.

Flintham, M. (2005):. Painting the town red: configuring location-based games by colouring maps. In *ACE '05: Proceedings of the 2005 ACM SIGCHI International Conference on Advances in computer entertainment technology*, pages 9–18, New York, NY, USA. ACM Press.

Fruh, C. and Zakhor, A. (2003): Constructing 3d city models by merging aerial and ground views. *IEEE Computer Graphics and Applications*, 23(6):52–61.

Google (2007): Pixagogo -photo maps. http://pixagogo.com. accessed on May 20th 2007.

Hightower, J. and Borriella, G. (2001): Location systems for ubiquitous computing. *IEEE Computer*, 34(8):57–66.

Hu, J., You, S., and Neumann, U. (2003): Approaches to large-scale urban modeling. *IEEE Computer Graphics and Applications*, 23(6):62–69.

Hull, R., Clayton, B., and Melamed, T. (2004):. Rapid authoring of mediascapes. In *Ubicomp*, pages 125–142.

LaMarca, A., Chawathe, Y., Consolvo, S., Hightower, J., Smith, I., Scott, J., Sohn, T., Howard, J., Hughes, J., Potter, F., Tabert, J., Powledge, P., Borriello, G., and Schilit, B. (2005): Place lab: Device positioning using radio beacons in the wild. In *Proceedings of PERVASIVE 2005, Third International Conference on Pervasive Computing*, Munich, Germany.

NavMan (2007): Navpix -navigation with pictures. http://www.navman.com/navpix. accessed on May 20th 2007.

NGA (2007). National geographical intelligence agency. http://earthinfo.nga.mil/GandG/wgs84. accessed on May 20th 2007.

Ordinance-Survey (2007): Osgb coordinate system. http://www.ordnancesurvey.co.uk/oswebsite. accessed on May 20th 2007.

SOMA (2006): Urban tapestries. http://urbantapestries.net. accessed on May 14th 2006.

Steed, A., MacColl, I., Randell, C., Brown, B., Chalmers, M., and Greenhalgh, C. (2004). Models of space in a mixed-reality system. *Information Visualisation 2004*, pages 768–777.

TomTom (2006): Tomtom -portable gps navigation for cars. http://www.tomtom.com. accessed on May 14th 2006.

15 Mobile Location-Based Gaming

Volker PAELKE[1], Leif OPPERMANN[2] , Christian REIMANN[3]

[1] Department of Cartography and Geoinformatics, University of Hanover
[2] Mixed Reality Laboratory, University of Nottingham
[3] C-LAB, University of Paderborn and Siemens AG

Abstract. In this chapter we provide an overview of the area of mobile location-based gaming and its relation to maps. Digital maps of various forms are essential to enable the implementation of mobile location-based games and support various key tasks from content authoring and interaction scripting over game state management and location-based interaction at run-time, to content presentation and user orientation. Based on an analysis of early experiments and currently existing systems we examine requirements with regards to hardware, software, standards and game design and discuss their implications for future developments.

15.1 Introduction

Mobile geoinformatics technologies for positioning and location recognition together with advances in mobile computing and wireless communication technology enable not only productivity applications but also the creation of new styles of mobile games. This includes games and entertainment experiences that are not only suited to mobile use but also exploit the dynamic location of the user in a meaningful way. In this chapter we examine mobile location-based games: mobile games in the more specific sense that exploit spatial location as part of the game design and thus "take the gaming experience out into the real world" (Benford, Magerkurth et al., 2005). Conventional video-games that are just delivered on a mobile platform (e.g. games on mobile phones, PDAs or Gameboys) are not considered.

15.1.1 Motivation

While spatial aspects and the task of "navigation" have been present in computer games from the beginning, this has mostly been in the form of simulation on a fixed display. Currently emerging technologies allow to go beyond this and enable the use of real world geographic environments for and within games. Within the domain of location-based services (LBS) such games form a promising market for early adopters in which development costs can be recuperated despite the technological limitations of current devices and networks. The successful development of mobile location-based games requires adequate base technologies, suitable development and management tools as well as appropriate game concepts.

Work on mobile gaming has initially focused on techniques to deal with the inherent constraints of mobile devices, namely small displays and limited interaction modalities. More recently, mobile gaming has started to explore the physical movement of players for gaming, largely driven by technology. A simple example for this is "geocaching", in which GPS receivers are used to find "geocaches": small waterproof containers containing a logbook whose positions are published on websites. More complex experiments with mobile games that exploit positioning technology include SingTel's *Gunslingers* and *Can you see me now?*. While these games still limit the use of the player's real-world context to positioning, the potential of the available technology is much larger: Large real-world geographic environments could be used as "game areas" (as large as the whole world in the case of geocaching) where the real-world is incorporated into the game content. In addition to this, user actions besides physical locomotion can be captured and used as interactions within games. The implementation of such game concepts relies on digital maps at multiple levels: as part of the game content, to specify location-based game content and behaviours during authoring, to handle spatially located multi-user interaction during game play and to provide users with adequate feedback on the current game state and their position within the real-world environment.

The mobile context of use differs significantly from the traditional use of computer games not only in the available devices but also in the characteristics of the environment and most importantly in the user's activities (Dey, 2001). Since the user is on the move the context can change continously. If context is to be exploited the game design has to ensure that the game remains not only playable but also enjoyable within a changing and only partly controllable environment. An import area in research on human-computer interaction for pervasive and ubiquitous applications is therefore dealing with the adaptation of user interfaces to changing context. One platform for this area is the mobileHCI Workshop series, which started in 1999 and will take place in Singapore in 2007[6]. Based on work for mobile and pervasive interfaces in general (Abowd and Mynatt, 2000) the specific challenges of game interfaces have become a subject of particular interest in recent years, as examplified by the large EU funded IPerG research project (Waern, Benford et al., 2004). Major research threads in the pervasive HCI domain include sensor technologies (Mandryk and Stanley, 2004) and interaction techniques (Cheok, Goh et al., 2004) as well as the usability of such interfaces (Jegers, 2004). Magerkurth et al. (Magerkurth, Chen et al., 2005) provide an overview of the large scope of pervasive game interfaces ranging from augmented tabletop games to the kind of mobile location-based gaming experiences discussed in this paper.

In the following sections we analyse the requirements of mobile location-based games with regards to game concepts, interaction techniques, hardware and software.

[6] http://www.mobilehci2007.org/

We then discuss which aspects can be adressed by different uses of digital maps and identify implications for future research, development and standardization activities.

15.1.2 Overview and relation to maps

While the basic technological problems faced in mobile applications are far from solved, the development has reached a stage at which many concepts can be successfully implemented in real-world applications. This is especially true in the gaming and entertainment domain where precision and reliability are often only of secondary concern and existing restrictions can be mitigated by clever game design. Maps are a central element of most location-based service (LBS) systems and form a central element of most mobile gaming applications. In the following section we discuss diverse uses of maps in mobile location-based games. These range from the content creation phase, over the technical implementation of the game to the interface with which a user is interacting. As a prerequisite it is useful to examine the basic components of mobile location-based gaming applications.

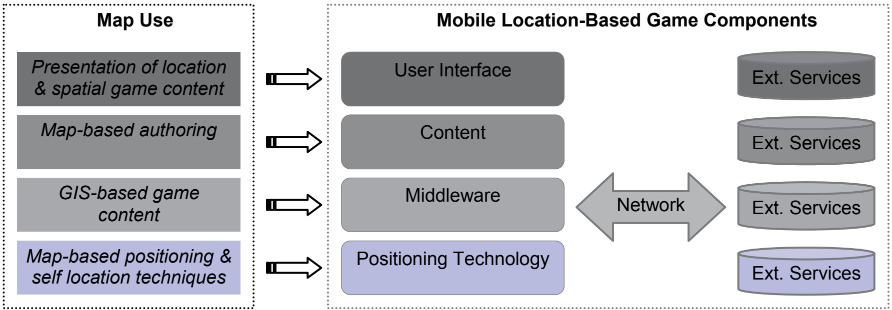


Fig. 15.1. Example layer model of a location-based game and associated functions of maps

Fig. 15.1 shows a layered view of typical mobile location-based game architectures. The foundation for any mobile location-based game as defined in this chapter is an adequate positioning technology as a prerequisite for the inclusion of the user's location into the game content.

The second level consists of middleware that manages content access and distribution. This can be as limited as data management on a local device or as complex as the management of distributed systems incorporating communication services with other devices or external servers. The third layer is the content of the application. A game-specific aspect of content is the need for active authoring (in contrast to direct information access in some other LBS incarnations) that is required to create meaningful game experiences. The top layer is the user interface. Again the user interface of mobile location-based games typically goes beyond the matter-of-fact content presentation in most LBS systems in order to create an engaging and compelling game experience. Consequently, the following subjects are of key relevance to the implementation of mobile location-based games:

- user interaction (content presentation and interaction)
- content authoring and acquisition
- content representation and management
- positioning and communication

An additional aspect is the hardware on which a mobile game is operating. The limited capabilities of current mobile devices impose serious constraints on the implementation of games and must therefore be considered throughout design, development and deployment of games.

Electronic maps of various forms play an important role in all stages of the game development and use cycle. In general a map provides a representation of objects and features in space. Since the introduction of electronic map implementations the traditional view of maps as a scaled and generalized 2D represention of the earth's surface has been expanded to include non-traditional uses, e.g. the use of electronic maps in route planning systems. Also, in the entertainment context artificial "maps" have a long tradition of use as the spatial representation of the game space, e.g. in board games. Given the list of subjects shown above, maps are used in the following functions:

User interaction: Closest to traditional maps and with a large overlap to maps in location-based services, visual map representations are commonly used as the base layer for information presentation in the user interface which is then augmented and enhanced with additional features that represent game content, players and opponents, or interaction elements. The visual presentation of the map in the user interface can range from traditional map styles over artificial game styles towards augmented reality styles in which the map content is visually integrated into the real-world surroundings of the user.

Content authoring and acquisition: In many location-based mobile games the acquisition of real-world environments and its integration with virtual media and content elements are key activities within the authoring phase of the development process. Maps are used not only as an appropriate metaphor for spatially aware authoring tools but electronic maps also provide the technological foundation for the implementation of such tools. Existing mapping tools and geographic information systems (GIS) are commonly used within the development process to support authoring and acquisition activities.

Content representation and management: While content authoring and acquistion refer to design-time activities the content representation and management covers the run-time handling of the game content, e.g. the handling of spatial queries. Again existing electronic map tools and GIS provide a foundation for the implementation of such functionality. To address the specific requirements of games (e.g. handling of group interaction, handling of latencies in sensor data or through network communication) and to adapt to the ressource limited mobile devices typically extensions or new developments are required.

Positioning and communication: Many positioning technologies used in mobile location-based games rely on (non-visual) maps (e.g. maps of network nodes and electro magnetic field strength maps in wireless communication networks) to determine the position of the user. While these maps typically remain invisible to the end-user they form an important information tool for developers and operate on similar principles and functions as more conventional maps. In some incarnations visual maps are used explicitly in the positioning task, e.g. in games that require interactive self positioning of the user.

The following sections provide a detailed view of the various (map-based) tools and technologies used within mobile location-based game development and deployment.

15.2 Review of exemplary mobile location-based games

Mobile location-based games are viewed as one of the future key markets for commercial location-based services. A key aspect is that mobile location-based games, especially those operating on common mass-market devices like smartphones or generic GPS receivers, have the potential to reach an audience beyond the traditional PC or console gamer. This includes activities like “geocaching” and its various "treasure hunt" extensions. Other game concepts require specialized devices that are often only available as prototypes in a research setting or require special infrastructure or operators that can currently not be maintained permanently on a financially viable basis. From a pragmatic technological perspective it is therefore useful to distinguish games that operate on current "standard" hardware, including all current commercial games, from those that require special features or infrastructure. This second category explores the available design space for future developments. Here two major approaches can be distinguished: One consists of games that explore new device or interaction concepts within a research setting, using specialized hardware or software. The second approach focuses on possible future game concepts and experiences and often takes the form of special events in which the necessary infrastructure and services are made available to a limited audience for a limited period of time.

For the following discussion we have adopted this distinction into "commercial", "event-based" and "research" games. With "commercial" we refer to games that operate on currently available standard hardware and could thus be marketed, "event-based" refers to those that require special infrastructure or operators for a limited time and are typically used to explore new game concepts with large audiences. The "research" category covers those games that rely on new hardware and software features that might become integrated into future gaming devices, but are currently restricted to laboratory settings.

15.2.1 Commercial games

The first commercial mobile location-based games were launched in 2000 / 2001. Since then quite a variety of location-based games were developed. The following examples represent some archetypical game designs. Additional up-to-date information can be found in blogs such as IN-duce[7], Pasta and Vinegar[8] or 7.5th Floor[9].

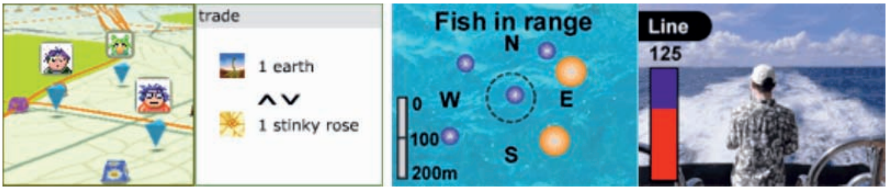


Fig. 15.2. Commercial games: *Mogi* (left 2 images), *Swordfish* (right 2 images)

15.2.1.1 Example 1: Mogi

Mogi [10] is a location-based collecting game from Japan, which is on the market since 2003. Using his mobile phone the player can pick up virtual items which are placed all over Japan. For every item the player gets a specified amount of points varying from item to item. Certain sets of items form groups, which give additional points when all items are collected. Players can also trade items with other players (see Fig. 15.2, left) to get more points for example by completing groups. Mogi uses GPS and the current network cell ID for positioning.

15.2.1.2 Example 2: GunSlingers

GunSlingers [11] is a shooting game from Singapore. The players can duel with other players nearby. The winner earns points and credits, which can be used to buy new equipment (like guns, armory, med-packs, etc). The game also features a mission mode in which the player has to transport items or assassinate virtual players. The game uses an operator-based positioning service which triangulates the players' position from the received signal strengths of his mobile phone at the surrounding (geo-coded) mobile phone masts. This method is said to achieve an accuracy of about 50 meters but based on the authors own experiences it can be expected to perform at least an order of magnitude worse on average.

[7] http://www.in-duce.net/archives/locationbased_mobile_phone_games.php
[8] http://tecfa.unige.ch/perso/staf/nova/blog/
[9] http://www.girardin.org/fabien/blog/
[10] http://www.mogimogi.com
[11] http://guns.mikoishi.com/gunsSingTel/index.html

15.2.1.3 Example 3: Swordfish

Swordfish [12] is a location-based fishing game that uses GPS to put the player in the game environment. The player has a virtual scanning device that visualises fishes in his direct vicinity on an abstract 2D map. The player then needs to move to the physical location of the fish in order to hook it. Once a fish got hooked, the game switches to a traditional arcade-style fishing game where the player has to reel in the fish using his phones' keypad up and down buttons to control the tension of the line without snapping it (see Fig. 15.2, right).

15.2.2 Event-based games

Event-based games generally utilize state of the art hardware which is not yet widespread amongst the public at that point in time. The game masters therefore have to supply all necessary infrastructure to the players. As a consequence, event-based games usually take place for a limited period of time and are to be played by a limited number of players. The time constraint causes most of the games to be staged during some host events, e.g. art festivals or jubilees.

Fig. 15.3. A *CYSMN* runner checking his PDA for nearby online players (left). An online player (white silhouette) in proximity to a runner in the 3D virtual world (middle). A player of *Riot! 1831* wearing headphones (right)

15.2.2.1 Example 1: Can You See Me Now?

Can You See Me Now? (*CYSMN*) is a game of chase in which online participants are chased through a 3D virtual model of a city by performers who, equipped with PDAs and GPS units, have to run through the actual city streets in order to catch them (Fig. 15.3, left). The performer's GPS positions and the online player's positions in the 3D virtual world are mapped into the same virtual coordinate system (Fig. 15.3, middle). It is considered a catch when a performer comes into 5 m proximity of an online player. The performer then takes a photo of the location and the game session of the online player is over. Players can send text-messages during the game and listen to the live-audio stream of the runners on the streets.

[12] http://www.blisterent.com/swordfish/index.jsp

CYSMN was first shown at the b.tv in Sheffield in the UK in December 2001. Since then it has toured to different art festivals across Europe and as far as Tokyo in 2005. The event usually lasts a few days with a couple of hours play-time each day.

15.2.2.2 Example 2: Riot! 1831

Riot! 1831 is an interactive play that was staged at the Queens Square in Bristol for a period of three weeks in spring 2004. It takes place on the same square where the Bristol riots of 1831 occurred. The riots were one of the most dramatic episodes in the history of the United Kingdom and caused great damage to the city, injuring or killing many of its inhabitants.

Players of *Riot! 1831* experienced a location-based audio play which let them immerse into the past and listen to fictional anecdotes that are based on the real events. The game audio-content is triggered by the GPS position of the player and delivered by a PDA through a set of headphones (Fig. 15.3, right).

15.2.3 Research games

Research games differ from commercial and event-based games. Typically, they are more focused on specific aspects and usually in a much earlier stage of development. By design they can concentrate on gaining knowledge about specific subproblems and do not necessarily have to provide a coherent and fun game (which is clearly the focus of the other types). Research games are usually closely monitored by scientists in order to capture as much knowledge as possible about the game-mechanics and the user interaction. The following examples can show only a small fraction of the wide variety of research games.

15.2.3.1 Example 1: Epidemic Menace

Epidemic Menace is a cross-media game which means that it is played using different gaming interfaces on different devices. It is build upon the story of a mad scientist contaminating a university campus with life-threatening viruses. The players arrive on site as teams of “medical experts” whose mission is to locate, capture, analyze and destroy those viruses using the different interfaces. As in the other examples the position of the outdoor players is determined using GPS. Indoor players use this information on map displays to assist outdoor players in their tasks. The presentation of the content is dependant on the current device. The audio-interface allows hearing the different virus types when the player is in proximity and captures them eventually. The stationary indoor interface gives an overview of the game-state including all mobile players and viruses. It allows to guide the mobile players and to analyze the viruses that have been captured.

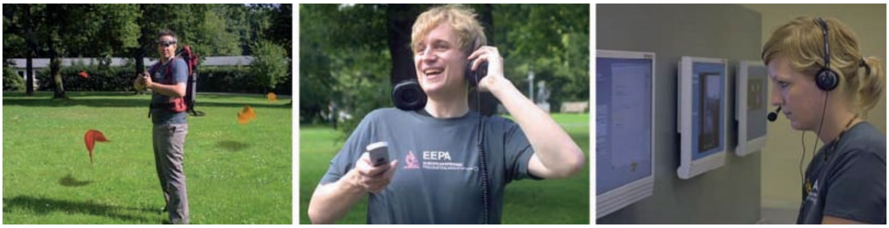


Fig. 15.4. An AR player surrounded by virtual viruses (left), an audio-player captured a virus (middle), a stationary indoor player guiding the mobile players (right)

The prototype of *Epidemic Menace* (Fig. 15.4) was first played on the Fraunhofer campus in Sankt Augustin in Germany on August 24th and 25th of 2005.

15.2.3.2 Example 2: Forgotten Valley

Forgotten Valley (Fig. 15.5, left) is a mixed-reality adventure game for PDAs, which allows the user to switch between conventional gameplay and the location-based mode. At the beginning of the adventure the player finds his avatar placed in the middle of an unknown map, not knowing where he or she is or how he or she got here. In location-based mode the user can start physically anywhere on the university campus in Paderborn, Germany, which is the real-world “game board” for Forgotten Valley.

In conventional mode the user can use the pointing device to move across the map. Exploring the surroundings in this manner, the player encounters different places where he or she may find hints about his whereabouts and how to move on in the game. In mixed-reality mode the user physically walks around on the university campus to discover the places relevant for the game.

The game uses GPS for positioning during navigation time, when the player moves to the next location. For the interaction at special location (e.g. the “Oracle”) fiducial markers and a camera-based tracking system are used to provide precise spatial information.

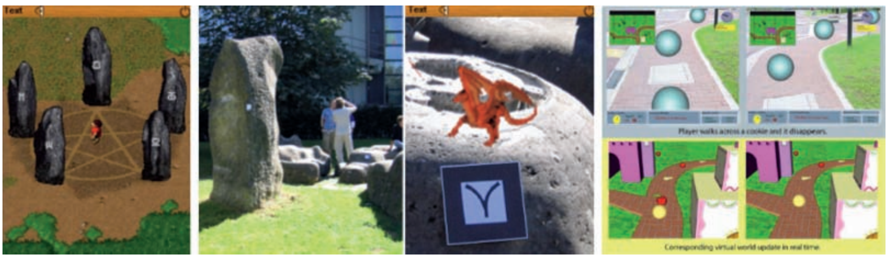


Fig. 15.5. *Forgotten Valley* in conventional and mixed reality mode (left 3 pictures), *Human Pacman* in mixed reality and virtual mode (right picture)

15.2.3.3 Example 3: Human Pacman

Human Pacman (Fig. 15.5, right) is a mixed-reality version of the traditional pacman game developed in Singapore. The players are equipped with a complete mobile augmented reality system consisting of a backpack with laptop, sensors, battery, etc. Head-mounted video-see-through displays allow to blend virtual objects into the field of view of players. Each player, who can be either pacman or a ghost, can see the other players and the typical yellow spheres on the way in front of him. By moving around the pacman-player can collect the spheres and gather points. The game uses GPS for positioning and additional sensors like gyroscopes to obtain the viewing direction of the player.

15.2.4 Summary of example games

The examples show a transition in the realized game concepts due to the involved technology, especially for positioning and display. The commercial games all use very basic hardware (mobile phones, sometimes with additional GPS) to maximize their market. This decision of course limits the realizable game concepts. So typically the following concepts are used in different flavours and combinations:

- **Navigation:** The player's physical movement is used to place the player in the game world. This is often supported by a map view like the runners' interface in CYSMN or a radar-like visualisation as in *Swordfish.* Position and proximity are also intrinsic parameters to the game-logic which are not revealed to the player through a map interface but possibly by other means (e.g. through sounds like in *Riot! 1831*).
- **Trigger**: The game-logic can trigger location-specific actions based on the players' position or proximity. The actions could be anything including playing sounds (*Riot! 1831*), messaging between players (*Mogi*) or altering the game-state, e.g. a catch in a multi-player environment (*CYSMN*) or a switch to a traditional single player game (*Swordfish*).
- **Collection**: Collection is a frequently used trigger-based concept. Virtual items, placed in the real world, can be picked up by position or proximity. Collection can be used as a key-feature (e.g. *Mogi*) of the game or as a side-feature (e.g. Medi-Pack in *GunSlingers*).

All these concepts are of course used by event-based and research games as well. The situation in an event-based game allows the use of non-standard or unconventional hardware, as there is no need to access a broad mass-market of devices. This can result in game designs, which are unrealizable otherwise (e.g. *Can you see me now?*). Having no market restrictions, research games can go even further and use a wide variety of additional sensors (e.g. gyroscopes) as well as displays (e.g. head-mounted displays). This allows not only integrating the above mentioned concepts into the game, but also using more sophisticated techniques for visualisation and interaction (e.g. augmented reality). Especially the divide between virtual and real world, which can be perceived very clearly by the triggering of arcade-style game, can be bridged this way, putting highly interactive parts of the game into the real world.

15.3 Mobile location-based game components

The following section provides a brief survey of technology components that are currently available for mobile location-based games. A common feature shared by all mobile applications is the need for small and lightweight devices, which in turn imposes constraints on the possible size (esp. relevant for displays), power supply and processing power. Key aspects to consider are available displays for information presentation, available interaction modalities, supported means of positioning and networking capabilities to establish connectivity.

15.3.1 Positioning

Making use of the users' real world location is a key feature for mobile location-based games that is not present in conventional game applications. Position is expressed in either absolute or relative coordinates. Possible technologies to obtain the users' position range from GPS in outdoor applications, over positioning in wireless networks and proximity to radio beacons to computer vision.

GPS-receivers deliver an absolute coordinate which is quite accurate on a global scale and gives a position update about once per second. No preparations are required to make use of GPS but one has to be aware that GPS does not work inside buildings as it requires a good, broad view to the sky.

Self-reported positioning (Benford, Seager et al., 2004) is an interactive process with the user. It requires a map-interface where the user can explicitly declare his position. This technique assumes that the user knows where he or she is so it wouldn't support him in cases of disorientation. When suitable, self-reported positioning can work quite well to deliver an absolute or relative position (near landmark) in a slow paced game.

A number of techniques allow a position to be automatically determined relative to known landmarks of different kinds. All of them require pre-mapping, i.e. giving those landmarks a coordinate in the first instance so that their position can be obtained when the player is close to them.

Wi-Fi positioning (e.g. Ekahau[13]) uses the ever-growing wireless local area networks (WLAN) which is especially wide-spread in city centres. By geo-coding WLAN access points in the pre-mapping phase and storing this information in an accessible way, one can obtain a relative or an absolute position reading using the

[13] http://www.ekahau.com/

hardware available in most laptops and PDAs. A key advantage over GPS is that Wi-Fi positioning works indoors as well.

Cell ID positioning is the equivalent for cellular mobile phone networks. In its basic incarnation it can deliver a rough position. The advantage of this technique is the very high coverage of the mobile phone networks which is basically available everywhere. The downside is that this technique is only supported by a subset of the phones, most notably on Symbian and Windows Mobile Smartphones. This limitation can be overcome by utilizing **operator-based positioning** where the mobile phone operator measures the proximity of a given mobile phone to the networks antennas. These proximities are then triangulated to result in an absolute position reading of better accuracy. The disadvantage of operator-based positioning is that each position reading has to be paid for which makes it currently too expensive for extensive use. It is also not available on every network.

Bluetooth devices and **RFID** tags can also be used to derive position information. Due to their ubiquity and lightness they are bound to be frequently moved around. This has to be considered when designing a Bluetooth or RFID positioning system. An absolute positioninig system could be developed by geo-coding only immobile devices and tags. A relative positioning system could sense the proximity of devices in respect to each other, without knowing an absolute position. Probably the most likely use-case for these mobile tags is to put them on the user and have the environment sense the user's proximity. A recent example for this approach can be found in a theme park in Britain: its omnipresent camera surveillance network is used to track and film individuals during their day out and offer them a personalized movie of their theme park visit as they leave[14].

Table 15.1. Classification of positioning techniques for Mobile Location-Based Games

	Precision	Update frequency	Indoor coverage	Outdoor coverage	Preparation required
GPS	High	High	-	High	None
Self reported (interactive)	Medium	Low	High	High	Map-Interface
Wi-Fi	Medium	High	Medium	Medium	Pre-Mapping
Mobile Phone, Cell ID	Low	Medium	High	High	Pre-Mapping
Mobile Phone, Operator-based	Medium	Low	High	High	None
Bluetooth	Medium	Medium	Medium	Low	Pre-Mapping
RFID	Medium	High	Medium	Low	Pre-Mapping
Computer Vision	Very High	Very High	Low	Low	Pre-Mapping

Precision	Very High (~ 1 cm) > High (~ 10 m) > Medium (~ 100 m) > Low (~ 100 m – 5 km)
Update frequency	Low (infrequent) < Medium (several seconds – minutes) < High (~ once a second) < Very High (more than 10 times per second)

[14] http://news.bbc.co.uk/2/hi/uk_news/england/staffordshire/4911916.stm

As a last example, **computer vision** techniques can be used to determine the player's position relative to landmarks like feature points or optical markers. This is a very sophisticated approach which can work well for some tasks that require high precision positioning information for optical augmentation.

Table 15.1 summarizes the presented positioning techniques and evaluates some key attributes on a rough scale. It can serve as an indicator of which technology might be suitable for a task depending on the tasks most important attributes.

15.3.2 Connectivity

Connectivity is a key feature in many mobile location-based game concepts and essential for applications in which multiple mobile users are expected to interact within the same game. While in desktop settings a relatively cheap and stable high-bandwidth network connection can be assumed the various technical and monetary constraints imposed by the possible networking techniques ranging from commercial networks (GPRS, UMTS) over WLAN hotspots to ad-hoc networks using local connections (e.g. Bluetooth, IR) must be considered in game development.

15.3.3 User interface

Displays are of central relevance as visual output is the most salient element of most game experiences. Typical options for displays include: Small screen displays (common in smartphones, PDAs and current portable gaming devices like the PSP), head-mounted displays (HMD, mostly in research settings) and mobile projectors (also mostly in research settings). Central constraints with regards to displays in mobile game platforms include the following:

The **limited resolution** of the graphics display of mobile displays is a key constraint that must be addressed by the game designers. Typical resolutions range from 100*80 pixels for mobile phones to 480*640 for PDAs and slightly higher resolutions in expensive head-mounted displays. These values have to be compared against mega-pixel displays common in current desktop environments.

The **small display size** of mobile devices is another central constraint, especially for applications that are aimed at a diverse user population, where possible vision problems must be considered. In most mobile game settings there is an upper limit for the possible size of the device (possibly in the range of current PDAs). New technologies like flexible OLED displays that could address these constraints are still a long way from mass-market applications, while alternative approaches (e.g. mobile projection) are restricted to special settings or events.

Many mobile devices feature only a **limited range of available colours**, e.g. several thousand colours on current PDAs compared to "true-colour" displays in desktop applications that can display millions of colours. In an outdoor mobile use context the range of distinguishable colours is often further reduced by environmental factors, e.g. lighting.

Table 15.2. Classification of typical devices for Mobile location-based Games

	GPS Device	Smartphone	PDA	Tablet PC	Laptop PC	HMD (+PC)
Common examples	Garmin Etrex, Haicom 406-BT	Nokia 6680, HTC Smartphone	HP iPaq, Palm Tungsten	Toshiba Portégé M200, HP TC	Dell Inspiron, Apple iBook	Saab AddVisor, Sony Glastron
Typical operating systems	Custom	Symbian OS (S60), Windows Mobile	Windows Mobile, Palm OS	Windows XP Tablet	Windows XP, Mac OS	(PC)
Typical Interfaces	Bluetooth, USB, Serial	Bluetooth, Infrared, Network	Bluetooth, USB, Network	Bluetooth, USB, Network	Bluetooth, USB, Network	VGA
Display size and resolution	2", 160 x 200 or none (LED)	2" – 4", 160 x 200 to 640 x 200	3,5", 240 x 320 to 480 x 640	12", 1024 x 768	12", 1024 x 768 to 15.4" 1680 x 1050	HMD, 320 x 200 to 1280 x 1024
Typical operating time	3 – 8 h	36 h	4 – 16 h	3 – 6 h	2 – 6 h	Ext. power supply
Average weight	50 – 200g	100 – 200g	140 – 200g	1 – 2.5 kg	2 – 3 kg	400 g – 1 kg
Approximate Price (2007)	100 – 300 EUR	200 – 500 EUR	200 – 700 EUR	2000 – 4000 EUR	1000 – 4000 EUR	300 – 60000 EUR
Common positioning techniques	GPS	(int.) / ext. GPS, Cell ID, (WLAN)	int. / ext. GPS, WLAN, (Cell ID)	ext. GPS, WLAN, other sensors	ext. GPS, WLAN, other sensors	(PC)
Interaction mechanisms	Keys	Keys, Joystick	Stylus, Keys, Joystick	Stylus, Keys, Joystick, ext. IO devices	Keyboard, Touchpad, ext. IO devices	Tracker (PC), Data Gloves (PC)
Programming languages	-	Java, C/C++, C#, Python, Basic	C/C++, C#, Basic, Java, Python	All common	All common	-

The interaction mechanisms and input devices offered by mobile devices also differ significantly from the conventional gaming domain. Typical options include a limited number of keys (smartphones, PDAs), small joysticks (smartphones, PDAs), touchscreen with pen (PDAs), positioning (e.g GPS), limited voice input, cameras as well as various device specific sensors. Key differences to conventional gaming thus include:

No standardization: While desktop applications can typically rely on the presence of a number of standardized input devices (e.g. joypads, keyboards and a 2D pointing device) the interaction facilities provided by mobile devices are device specific. While the lack of a full alpha-numeric keyboard in typical platforms is less of a problem in mobile games, the diverse sets of available key layouts can be problematic. It is also typically impossible to extend mobile devices with task specific interaction modalities.

No mouse: Most desktop applications rely on the use of a 2D pointing device (typically the mouse, but also track-pads etc.) as the main interaction mechanism. Pointing devices on mobile devices often have significantly lower resolution (e.g. touch-screens) or require the use of additional hand-held components (e.g. pens on PDAs). Many mobile devices provide no 2D pointing mechanism at all (e.g. mobile phones).

Specific interaction techniques: Most mobile devices provide additional input mechanisms that are not applicable in a static desktop setting (e.g. location sensing, camera-based input). Experience with these techniques is currently very limited, making their systematic use in user interfaces difficult.

15.3.4 Spatial interaction

Position information can be exploited to trigger interactions. While position-based interaction for single players is well studied in location-based services, interesting interactions can be achieved by integrated analysis of multi-player location and movement information. The analysis of group movement pattern is a recent research topic with high potential. Mobile location-based games are especially suited as a test and evaluation platform for this type of interaction because of the access to large user bases and the less strict precision and reliability requirements compared to productivity applications. The following table provides an overview of the kind of interaction events that can be derived from absolute or relative position or movement information, as it is provided by diverse positioning technologies. The table is split into the main columns "Single Player" and "Multi-Player".

Table 15.3. Single and multi-player position-based interaction techniques

	Single Player		Multi-Player	
	Position	Movement	Position	Movement
Absolute	POSITION	PATH, SPEED	MEET	PATH, SPEED, TRACKING, SEPARATING, CONVERGING
Relative	PROXIMITY	DIRECTION, SPEED	MEET	DIRECTION, SPEED, PATTERN

POSITION	trigger position dependant action
PROXIMITY	trigger proximity dependant action
PATH	trigger path dependant action, record path as completed
DIRECTION	direct parameter input, e.g. numeric value, direction
SPEED	direct parameter input

MEET	trigger position dependant action if n players are present
PATH	trigger path dependant action, record path as completed (only if n players are present and move in parallel)
TRACKING	detect if at least n players are following another (e.g. to trigger action that divert the following players, to adapt the difficulty for players accordingly)
SEPERATING	detect if a group of players splits up
CONVERGING	detect if n players are (likely) to meet in the future
SPEED	direct parameter input (only if >= n players present)
DIRECTION	direct parameter input, e.g. numeric value, direction
PATTERN	detect if a group applies a certain motion patterns (e.g. to trigger actions, gestures)

The detection of spatial group movement or position patterns can be tackled by pattern recognition algorithms. While many patterns (esp. in the single user case or for the detection of group actions within groups of fixed size) can be effectively detected using well known algorithms the detection of complex group interactions where interesting structures have to be detected in the large sets of position information is more difficult (Anders, 2004; Sester and Anders, 2005). Possible approaches include supervised or unsupervised classification algorithms (e.g. (Koperski and Han, 1995)). One promising approach uses a hierarchical parameter free clustering algorithm (Anders, 2004) based on neighbourhood graphs. The application of the approach to the mobile gaming context is an area of ongoing research. Future work will aim to extend the 3D patterns listed above (which expect more or less simultaneous actions of the users who form a group) to the handling of 4D cases in which a significant time delay (from minutes to days or even longer) exists between user actions. On the content side it will be interesting to explore possible game scenarios enabled by these kinds of interaction triggers.

15.3.5 Distributed infrastructure

As mentioned earlier, mobile location-based games are different from traditional games in that they utilize the player's context. In its simplest form this can be achieved by a sensor like a GPS receiver that inputs its data into a game played by a single player on a single machine. The built-in games of some Garmin GPS-receivers are good examples of such games. They can be played with just the GPS device and some free outside space to roam.

However, more often a mobile location-based game is designed to be played together by multiple players. This extension of scope introduces a lot of complexity to the internals of the game. Much of that complexity is due to the distributed nature of mobile devices involved. Games on stationary[15] devices can usually rely on a fast and low latency internet connection in order to work and would consequently just pop the user out of the game if the connection is temporarily down. Games for mobile network devices, on the contrary, have to cater for temporal disconnection. They also usually have to work with higher latency networks like GPRS, although the advent of UMTS and the ever increasing availability of public Wi-Fi access points will mitigate this problem in the future. In addition to disconnections and latency the synchronization of states across the distributed system and a persistent data-storage for all assets (including the log-files) remains a key issue. A mobile location-based game engine therefore has to tackle these problems in some way. This section presents possible solutions of different scale and scope. It also shows that no standard has emerged, yet.

[15] we regard wireless laptops in office settings to be stationary as well

15.3.6 Custom game-engines

Many mobile location-based games implement their own custom game engine by tying together available standard components for distributed applications and filling the gaps with custom code. A typical approach would incorporate a client-server architecture that communicates over HyperText Transfer Protocol (HTTP) using a standard Linux, Apache, MySQL, PHP/Perl/Python (LAMP) installation. The architecture in Fig. 15.6 is one such example which can be broken down into four main components:

1. mobile units, running the purpose built user interface, using a positioning technology and incorporating some kind of network communication
2. a global web server for data storage and retrieval
3. an alternative public interface to the data-storage, e.g. a private/community player-interface or a spectator client for non-players
4. an admin interface that allows modifications to the database

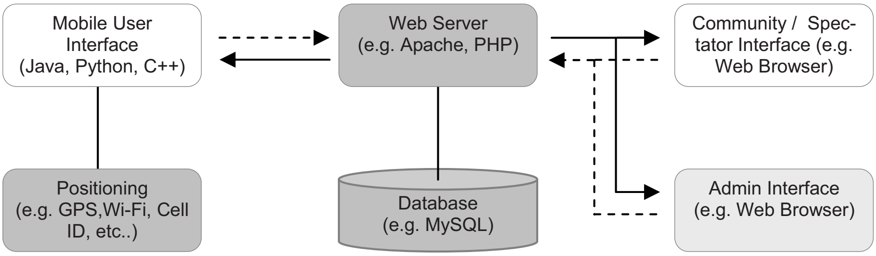


Fig. 15.6. Example architecture of a custom mobile location-based game using standard components

The communication between the components is done via HTTP requests (depicted as arrows in the Fig. 15.6) to the web server and its responses (the arrows in the opposite direction). The web server hosts dynamic pages (e.g. PHP) and a database for persistent data-storage. Due to the success of the World Wide Web, HTTP is available on almost every network connection. HTTP requests also pass most firewalls and proxies of corporate, educational and private networks since this is a requirement for web browsers to work. This makes it easy to deploy the public and admin interfaces based on HTTP and makes HTTP an almost ideal choice for mobile location-based games. The only downside is that HTTP doesn't allow messages to be pushed from the server to the client. Each communication has to be initiated by the client. Therefore the mobile clients constantly have to query the server for possible events. A number of approaches exist to overcome this limitation by implementing server-push (and more). The following is a selection of 3 example solutions that go beyond HTTP.

15.3.6.1 Elvin

Elvin (Segall and Arnold, 1997) is both a network communication protocol and middleware which can be used to build a custom, more flexible mobile location-based game-engine. It implements a content-based publisher–subscriber pattern in a client–server architecture. The server can push messages to the clients as they occur. A strength of Elvin is the content-based routing which means that the recipients of a message are determined based on the messages' content.

15.3.6.2 EQUIP

EQUIP (Greenhalgh, Izadi et al., 2001) is a framework for developing distributed interactive systems. It is a cross-platform, cross-language middleware for data-sharing between networked clients. EQUIP is based around the concept of a shared data-space, where producers publish information to a data-space and keep it up-to-date. Information consumers can use pattern-based techniques to register their information interests with the data-space and stay up-to-date with any changes through server pushed messages. A new version called EQUIP2 was under active development at the time of writing (early 2007)[16].

15.3.6.3 MUPE

The Multi-User Publishing Environment (MUPE [17]) is an Open Source application platform for creating mobile multi-user context-aware applications (i.e. mobile location-based games) on mobile phones. Similar to the previous examples it incorporates a client-server architecture where the clients are run on the mobile phones and the server is running in the Internet. It facilitates the content description and distribution by having a unified client-interpreter (written in Java) that subscribes to a central web-service which provides it with the XML-formatted game-content. The game-content just has to be defined on the server and gets automatically deployed to the clients (when requested by a client). Clients then start interpreting the game-content rules depending on the user's context (possibly including GPS position). MUPE does not yet have as sophisticated messaging mechanisms as Elvin or EQUIP, but it allows communication between clients as either a broadcast to all other clients or exclusive to a specific client. Messages are then routed via a server push. MUPE is also less general than Elvin or EQUIP in that it is limited to mobile phones. These limitations are used to make domain specific applications easier to develop. MUPE basically facilitates the communication and synchronization problem and allows connecting to a number of sensors on the client side, including GPS.

[16] http://www.mrl.nott.ac.uk/~cmg/EQUIP2/
[17] http://www.mupe.net/

15.4 Mobile location-based game tools

Much research and development work in mobile location-based games has so far focused on the creation and demonstration of new technologies with limited consideration of design and game play. The design and implementation of mobile location-based games on the programming level is a difficult and time consuming task that requires expertise in a wide range of constantly evolving technologies. A pure programming approach to mobile location-based games using standard APIs as the main abstraction to handle communication, positioning and interaction tends to create a gap between mobile technology experts on one side and (potential) content designers on the other.

Similar problems were previously encountered in the creation of multimedia applications and conventional desktop or console games. Two main approaches to the problem can be distinguished: The creation of high-level toolkits that present the required functionality at a higher level of abstraction and the creation of visual authoring tools that provide content designers with the means to approach game development from the content side.

Visual authoring tools are commonly used in combination with run-time game engines that control media delivery and interaction. If required these can be augmented by custom scripts or extensions. The development of appropriate authoring tools is a key to make the creation of mobile location-based games a viable venture. However, while several authoring tools for mobile location-based games have been developed in research projects there are few examples (if any) in which these have actually been applied by real content developers outside the project context in which they were developed. Experience with these tools suggests two possible reasons: A first group of existing tools only provides elaborate visual interfaces to the base software. Thereby they fail to support a content-centred way of work. A second group provides completely visual editors that in theory are suitable to support a content-centred workflow, but are in practice limited to the creation of simplistic toy applications in a single domain.

15.4.1 Requirements for authoring tools

Based on a pilot study we have identified requirements for flexible, extendable visual authoring tools. To be effective these must support an appropriate conceptual model and should be designed to operate within a structured design process that also integrates with other tools for media production and content management.

15.4.2 The mobile environment

The environment for mobile location-based games combines a real world setting (on which the designer has only very limited influence) with artifical game specific content. The constraints imposed by the environment can be severe and must be easily accessible and understandable to game designers during the authoring phase. Maps that provide a suitable representation of the real world context into which the game content is embedded are the most promising representation of the environment during the authoring phase. An interesting alternative is examined in experiments that also consider authoring on-site within the real physical environment (Piekarski and Thomas, 2001).

15.4.3 The goal of entertainment

The goal of a mobile location-based game is typically an abstract goal like the creation of entertainment and pleasure within the specific concept domain (game concept). It is sometimes combined with a skill acquistion or learning task. Such a goal is difficult to capture in a formal specification which makes it difficult to derive requirements by formal means. Instead typically design oriented approaches with close user involvement must be used. The task is further complicated by a lack of validated design expertise with all but the most basic mobile game concepts. These factors require a development process based on iterative refinement that should be reflected in the authoring tools by enabling early and repeated evaluation with minimal overhead. It is noteworthy that both event-based games discussed earlier (*CYSMN* and *Riot! 1831*) have been developed by engineers supplying the infrastructure and artists putting in the content. This duality is an iterative model in itself and apparently a good way of approaching the complex goal of entertainment.

15.4.4 Need for evaluation through use of prototypes

The lack of a formal specification and the limited design expertise make it difficult to apply techniques like expert reviews. Experimental evaluation through user tests thus remains as the main option. To limit the overhead of conducting (repeated) tests support for data logging should be integrated into the deployment plattform. Since games are typically used voluntarily an attractive and acceptable interface is a key requirement and this has to be ensured through tests. A coherent approach to mobile location-based game authoring must thus provide the following:

A user-centred design process that allows close collaboration between developers, designers and customers. Iterative design processes (e.g. based on ISO 13407) have emerged as the defacto standard in most domains that have to address informal specifications and novel technologies. The user-centred design process must match with a suitable conceptual model that allows the specification of game content on a suitable level of abstraction and is accessible to all stakeholders in the design process (e.g. developers, designers and customers). Furthermore, due to the openness of the game design problem and the current lack of expertise with many mobile location-based game design concepts early and repeated testing should match well with the choosen design process.

A conceptual model that fits the real world environment and allows to integrate game elements and group user interaction as dynamic and proactive content. Since conceptual models are used by designers, both to reason about a design problem and to communicate with customers, they should be familiar to both groups. Conceptual models based on maps are promising as maps constitute a well known base concept, provide the necessary means to access and specify spatial content and have a direct mapping to implementations, e.g. in GIS systems. Extensions to dynamic maps are required to handle dynamic moving objects and the handling of single user and group interactions.

A suitable and standardized implementation platform: If a mobile game is intended to be used beyond small-scale demonstrators the use of a standardized implementation platform as the base for the run-time environment is essential.

15.4.5 Authoring tools

Dedicated authoring tools and techniques for mobile games have begun to emerge. All examples produced to date share a common overarching approach: The idea of taking a map of the physical area in which the experience is to take place and then augment it with game content, typically by placing or drawing a series of regions, locales or hotspots on top of this that are triggered according to basic events such as participants enter and leave them.

The Mediascape tool from the Mobile Bristol project is a typical tool in which designers can place different sized triggers on a map of the gaming-area and associate these with a range of events (Hull, Clayton et al., 2004) (Fig. 15.7). Mediascape has been used to support a variety of location-aware applications including: *Scape the Hood* (Reid, Melamed et al., 2006) in which users explored an area of San Francisco, triggering stories as they entered different city blocks and *Riot! 1831* (Reid, Hull et al., 2005), an interactive audio-play which has been described in the examples section above.

A similar approach is the idea of Colour Maps, in which artists directly paint freeform location triggers in different colours onto a raster-image. Each unique colour is used as the key to identify the different events and the Colour Map (the artist's raster image) serves as a look up table for which events should be triggered according to the user's position. This affords the designers a high degree of flexibility in terms of the size, shape and layout of the trigger-zones and also enables them to use existing and familiar paint tools. This approach was used to support the projects *Uncle Roy All Around You* (Flintham, 2005) in which a group of artists defined a trail of text clues around a city for players to follow as part of an interactive performance and *Savannah* (Benford, Rowland et al., 2005), a game in which groups of children played being lions on a virtual savannah. The virtual savannah appeared overlaid on their school playing field.

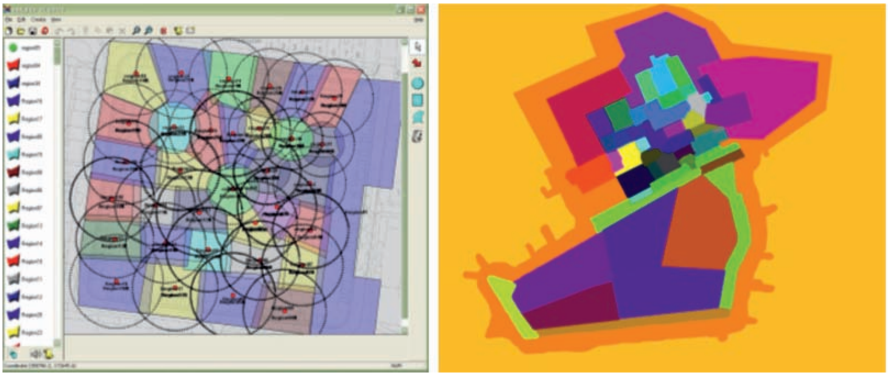


Fig. 15.7. Mobile Bristol Editor showing the authoring of *Riot! 1831* (left), a colour map showing the layout of *Uncle Roy All Around You* (right)

Similarly to the Mediascape tool, Caerus (Naismith, Sharples et al., 2005) is a tool designed to enable a designer to administer trigger areas overlaying a digital map. The user can add further maps, points of interest and multimedia streams. Caerus has initially been used to author a guided tour at the University of Birmingham's botanic garden.

15.4.6 Preparing to author

Authoring with any of the above mentioned tools requires at least a map of the area to author for. This background layer is usually acquired by:

- digitizing a paper map or an ortho-rectified (i.e. bird's eye with no skew) aerial photo, or
- accessing a digital map server

The acquired map needs to be geo-referenced by at least two points which have to be known in map-coordinates and real world coordinates. The most commonly used coordinate system for real world coordinates is the *World Geodetic System* (WGS84) which is a geoid coordinate system around the centre of the earth that can reference any point in, on and above the earth in latitude, longitude (both in degrees) and altitude (in meters). Other coordinate systems such as the German Grid or Ordnance Survey Great Britain (OSGB36) are in national use and can be utilized as well but they usually require a transformation from the standardized WGS84 system that is provided by every GPS device.

Additional information might be required for authoring. A lot of information like the road and transport networks, buildings and possibly problematic areas (e.g. in terms of GPS reception) can usually already be derived from a suitable map. Other features might not be represented in the map and have to be collected in some way. These features possibly include interesting places for game play or relevant infrastructure information. But they also include information about the players' behaviour and communication between players. It is therefore good practice to log potentially interesting and relevant information before and during game-events to create a richer picture of the environment. This also facilitates evaluation of the game for possibly adjustments during a game and improvements for future sessions.

15.5 Conclusions

In this chapter we have provided an overview of the current state of the art in mobile gaming and its relation to maps throughout the game development and use cycle. Current developments in mobile location-based games that explicitely exploit position information as part of the game content or as a means of interaction control are still largely in an experimental stage but demonstrate the wide potential of playing within large real-world environments: The close integration of real-world locations and physical objects into games allows to create interactive experiences that go beyond typical computer games. They emphasize physical actions and exercise, thus enhancing physical skills of the players and integrate coordinated group actions, thus training group and team work skills. By incorporating real world features, they also allow players to learn about the real environment as part of the game experience. Mobile location-based games have the potential to create experiences that are valuable and attractive to large audiences beyond traditional computer gamers. To exploit this potential further, research into base technologies, game concepts and tools is required. Central areas of future research therefore include:

- Interaction based on large scale physical movement within real-world environments
- Suitable mobile devices featuring sensors to capture the required information reliably
- Interaction techniques and metaphors for single-user and group interaction based on these
- Interaction in a real-world environment both outdoor and indoor, including blended interaction where the same techniques and devices are applied when moving from an indoor to an outdoor location and incorporating real-world objects and environments
- Appropriate multi-media feedback representations to ensure situation awareness in single and multi-user interaction scenarios
- Suitable authoring and development tools and processes that support the effective creation of mobile location-based game content

- Research into game concepts that explicitly exploit position information of players and real-world environments as part of the game content
- Suitable implementation tools and middleware, including efforts aimed at standardization and interoperability

Maps play a key role in these developments. On one side visual maps are a key element to provide both game developers and users with a presentation of location-based information that is adequate and relevant to their respective taks. Appropriate map styles and interaction techniques based thereon will be key elements to make both future authoring tools and the game interfaces themselves enjoyable and useful. On the other side, electronic maps form the backbone for the implementation of mobile location-based game functionality and location-based interaction techniques like group behaviour detection. Map-based tools and technologies will therefore be a key element for the successful development and deployment of future mobile location-based games.

Acknowlegements

We gratefully acknowledge the support of IPerG, the European Project on Pervasive Gaming, project FP6-004457 under the European Commission's IST Programme (www.pervasive-gaming.org). We are also grateful for support from the German Ministry of Education and Research (BMBF) in the D-LOMS project (www.dloms.org) as part of the ITEA project LOMS (www.loms-itea.org).

References

Abowd, G. D. and E. D. Mynatt (2000): "Charting past, present, and future research in ubiquitous computing." *ACM Transactions on Computer-Human Interaction* **7**(1): 29-58.

Anders, K.-H. (2004): Parameterfreies hierarchisches Graph-Clusteringverfahren zur Interpretation raumbezogener Daten. *Institut für Photogrammetrie, University of Stuttgart.*

Benford, S., C. Magerkurth, et al. (2005): "Bridging the Physical and Digital in Pervasive Gaming." *Communications of the ACM* **48**(3): 54-57.

Benford, S., D. Rowland, et al. (2005): *Life on the Edge: Supporting Collaboration in Location-Based Experiences.* CHI, Portland, Oregon, USA.

Benford, S., W. Seager, et al. (2004): *The Error of Our Ways: The Experience of Self-Reported Position in a Location-Based Game.* Ubicomp, Nottingham, UK, Springer.

Cheok, A. D., K. H. Goh, et al. (2004): "Human Pacman: A mobile, wide-area entertainment system based on physical, social, and ubiquitous computing." *Personal and Ubiquitous Computing* **8**(2): 71-81.

Dey, A. K. (2001): "Understanding and Using Context." *Personal and Ubiquitous Computing* **5**(1): 4 - 7.

Flintham, M. (2005): *Painting the Town Red: Configuring Location-Based Games by Colouring Maps*. Advances in Computer Entertainment Technology, Valencia, Spain.

Greenhalgh, C., S. Izadi, et al. (2001): The EQUIP Platform: Bringing Together Physical and Virtual Worlds, University of Nottingham.

Hull, R., B. Clayton, et al. (2004): *Rapid Authoring of Mediascapes.* Ubicomp, Nottingham, UK, Springer.

Jegers, K. (2004): *Usability of Pervasive Games.* PerGames Workshop, Vienna, Austria.

Koperski, K. and J. Han (1995): *Discovery of Spatial Association Rules in Geographic Information Databases*. Advances in Spatial Databases, Portland, Maine, USA, Springer.

Magerkurth, C., D. Chen, et al. (2005): "Pervasive Games: Bringing Computer Entertainment Back to the Real World." *ACM Computers in Entertainment* **3**(3).

Mandryk, R. and K. G. Stanley (2004): *Gemini: Accumulating Context for Play Applications.* Ubicomp Workshop on Playing With Sensors, Nottingham, UK.

Naismith, L., M. Sharples, et al. (2005): *Evaluation of CAERUS: a Context Aware Mobile Guide.* mLearn, Cape Town, South Africa.

Piekarski, W. and B. H. Thomas (2001): *Tinmith-Metro: New Outdoor Techniques for Creating City Models with an Augmented Reality Wearable Computer.* International Symposium on Wearable Computers, Zurich, Switzerland.

Reid, J., R. Hull, et al. (2005): *Magic Moments in situated mediascapes*. Advances in Computer Entertainment Technology, Valencia, Spain.

Reid, J., T. Melamed, et al. (2006): *Scape the Hood : A design case study of a location based digital story mediascape.* Digital Interactive Media Entertainment & Arts, Bangkok, Thailand.

Segall, B. and D. Arnold (1997): *Elvin has left the building: A publish/subscribe notification service with quenching*. AUUG, Brisbane, Australia.

Sester, M. and K.-H. Anders (2005): *Detection of Motion Patterns in Groups of Moving Sensors*. GI-Tage, Muenster, Germany.

Waern, A., S. Benford, et al. (2004): *IPerG Position Paper*. PerGames, Vienna, Austria.

16 Mobile Maps and More – Extending Location-Based Services with Multi-Criteria Decision Analysis

Claus RINNER

Department of Geography, Ryerson University, Canada

Abstract. Maps are often used as decision support tools in both, desktop geographic information systems (GIS) and mobile GIS environments. The decision support capabilities of current location-based services (LBS) are limited to navigation support and database querying with no analytic evaluation of the attractiveness of alternative destinations being offered. This chapter demonstrates how LBS can be extended with specific decision support functionality, namely multi-criteria decision analysis (MCDA). MCDA was recently transferred to the mobile GIS platform illustrating how LBS user preferences can be represented by the parameters in a MCDA method and will lead to personalized decision outcomes. An extension to a collaborative crisis management scenario is proposed, in which mobile decision-makers have MCDA tools at hand to help them make more informed choices. This chapter describes the scenario and derives a client/server architecture as well as the user interface and map design for a mobile decision support system for emergency response.

16.1 Introduction

In both, desktop geographic information systems (GIS) and mobile GIS applications, maps often serve as decision support tools. At present, the decision support capabilities of location-based services (LBS) are limited to navigation support and database querying. For example, an LBS may indicate a nearby point of interest to the mobile user and the shortest way to access it. However, the choice between different possible destinations will be left to the user, and no analytic evaluation of the attractiveness of the alternatives is being offered.

GIS have been described as generators for spatial decision support systems (SDSS), which are specialized applications providing decision support tools to decision-makers. Multi-criteria decision analysis (MCDA) methods form a group of such decision support tools. MCDA aggregates standardized attributes of decision alternatives into an evaluation score for each alternative, thus making it possible to rank alternatives based on their performance. MCDA was introduced to GIS in the context of site selection and land-use allocation in the 1990s (Malczewski, 1999).

MCDA concepts were recently transferred to the mobile GIS platform. Raubal & Rinner (2004) have shown that mobile user preferences can be represented by the parameters in a MCDA method and these will lead to personalized outcomes in mobile decision-making. Rinner & Raubal (2004) have extended the concept of personalized multi-criteria decision strategies using the ordered weighted averaging method that supports different approaches to decision risk and criterion trade-off. Finally, in Rinner et al. (2005), alternative user interface designs for location-based multi-criteria decision services were tested to simplify mobile decision-making processes.

Emergency situations are characterized by the need to make important decisions quickly and in collaborative situations. Cai et al. (2004) describe a hypothetical crisis management scenario, in which a first responder team uses mobile map-based communication with the emergency operations centre to find a shelter for hurricane victims. Here, an extension to this scenario is proposed, in which mobile decision-makers have multi-criteria tools at hand to help them make more informed choices. The parameters for decision-making could be shared between emergency teams to allow personal as well as group-default preferences.

After introductions to MCDA from a Geoinformatics perspective (16.2) and to location-based decision support using MCDA methods (16.3), the scenario is presented in detail (16.4) and a client/server architecture (16.5) as well as the user interface and map design (16.6) for a mobile decision support system for emergency response are derived. A make-up implementation using ESRI's ArcPad is also discussed and an outlook on further research provided (16.7).

16.2 Multi-criteria decision analysis in geoinformatics

Simon (1977) suggests a structure for analyzing human decision-making processes by distinguishing the *intelligence*, *design*, and *choice* phases. In the intelligence phase, a situation is examined for conditions calling for a decision. In the design phase, managers and planners develop alternative solutions to the decision problem identified in the intelligence phase. In the choice phase, decision-makers choose the best decision alternative. In the context of decision problems with a spatial connotation, Malczewski (1999) examines the potential for applying spatially enabled methods in Simon's decision phases. While the intelligence and design activities can mostly be covered by multi-purpose spatial analysis methods, the choice phase requires specific methods still missing in most GIS.

The choice phase is "what many people think of as making a decision" (Malczewski, 1999). It requires formal methods (decision rules) to select feasible alternatives and to rank them with respect to the decision-makers' preferences. As humans tend to base rational decisions on an assessment of multiple decision criteria, MCDA methods have become important tools in management sciences and operations research. By incorporating quantifiers for the decision-maker's preferences (e.g. criterion importance weights), these types of decision rules are capable of solving semi-structured decision problems.

GIS have been described as spatial decision support systems (SDSS) per se, or as generators for more specific SDSS (Keenan, 1997, Rinner, 2003). On a conceptual level, Malczewski (1999) defines the necessary components of an SDSS in the narrower sense as

- Geographic database
- Model base
- Dialog (user interface)

MCDA methods are a specific type of model within SDSS. Janssen & Rietveld (1990) and Carver (1991) were among the first to analyze the benefits and potential traps of integrating MCDA with GIS. Among the methods that have been used in conjunction with GIS are location-allocation algorithms (ArcInfo), linear programming, ideal point analysis (CommonGIS), weighted linear combination, and the analytic hierarchy process (Idrisi). Different integration strategies have been presented ranging from loose coupling (data transfer between GIS and decision support tools via files or user input) to full integration (implementation of decision rules in GIS macro language).

Fig. 16.1 presents a schematic workflow for GIS-based multi-attribute decision-making, the simpler type of MCDA procedures. First, decision-makers have to agree upon a set of feasible decision alternatives (e.g. possible destinations in the case of LBS). Next, evaluation criteria have to be selected, upon which a rational selection of an alternative can be based. The criteria have to be metric and standardized in order to make them commensurate. The relative importance of each criterion in the criterion set has to be quantified using criterion weights. A decision rule defines the way in which the standardized criterion scores are weighted and combined into an overall evaluation score for each decision alternative. Finally, from the evaluation scores, a ranking of alternatives can be derived and the top-ranked alternative represents the suggested decision. The decision process may also involve iterations in which the previously determined input parameters are revised.

In comparison to unstructured decision-making and to structured paper and pencil methods, GIS-based MCDA offers obvious benefits: The formal approach to establishing decision criteria and assessing alternatives on this basis allows for objectivity and reliability in a rational decision process. Alternatives can be screened efficiently in order to pre-select feasible options. Finally, the sensitivity of the outcomes can be analyzed, thus contributing to a *review* phase in an extended decision-making model.

On the other hand, rational decision-making in planning exercises has been criticized in general. Formal methods often cannot account for stakeholders' knowledge and intuition, and may result in a misleading reliance in decision outcomes. Also, the decision-makers' preferences often shift and remain unstable throughout a decision-making process. Critical technical issues include problem complexity due to the often large volume of spatial data. In practical use, preference input for MCDA methods is limited to a small number of decision criteria and weights may be difficult to elicit. Previous research also found that applying different MCDA methods to the same decision problem generates diverging results that leave decision-makers stunned (Heywood et al. 1995).

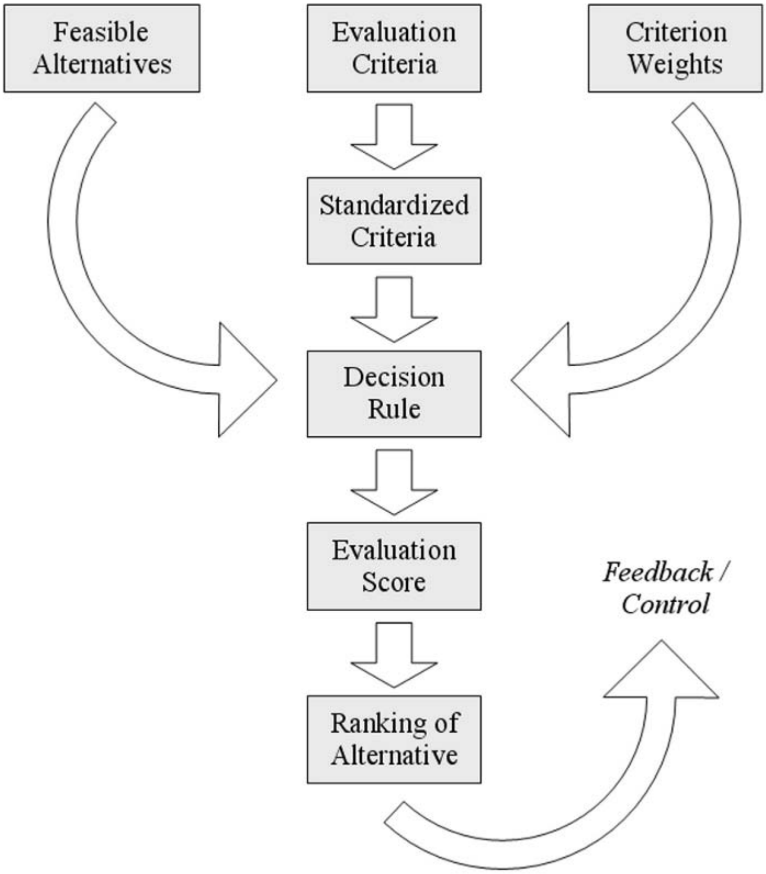


Fig. 16.1. Workflow for GIS-based multi-criteria decision analysis

To address inconsistency between evaluation results from different multi-criteria decision rules, decision support toolkits are being developed, which offer access to several methods for comparison of results. Models from other disciplines such as fuzzy set theory are also being adapted to spatial decision-making problems, e.g. the *ordered weighted averaging* (OWA) method (Yager, 1988; Jiang & Eastman, 2000; Rinner & Malczewski, 2002; Malczewski & Rinner, 2005).

Changes in the use of maps also influenced usage patterns for spatial decision support tools. DiBiase (1990) and MacEachren & Ganter (1990) were among the first to identify new map uses in the scientific process. They argued that increasingly, maps are not only used to present results of spatial analysis, but to explore spatial data and discover patterns, based on which scientific hypotheses can be developed. MacEachren (1994) introduced the cube model of map use in which he associates visualization with the use of interactive maps to reveal new insights into data. Parallel to the shift in map use from presentation to visualization, exploratory MCDA tools have been presented recently (Jankowski et al., 2001; Rinner & Malczewski, 2002;

Malczewski & Rinner, 2005; Rinner & Taranu, 2006). In addition, Rinner (in press) discusses the move from map-based exploration of spatial data to map-based exploration of the outcomes of spatial analysis processes such as MCDA processes.

16.3 Location-based decision support

Location-based services (LBS) assist people while they move through the physical environment. Current research on LBS addresses a variety of topics such as network architectures and information technology standards (Adams et al., 2003; Peng & Tsou, 2003; Ahn et al., 2004), positioning techniques and data collection (Mountain & Raper, 2001; Miller, 2003; Spinney, 2003), human-computer interface (Pundt & Brinkkötter-Runde, 2000; Meng, 2005), map design adaptation and personalization of services (Hjelm, 2002; Zipf, 2002a, 2002b, 2003; Gartner et al., 2003; Sarjakowski & Nivala, 2005), market opportunities and business cases for LBS (Beinat, 2001; Benson, 2001; Barnes, 2003), as well as locational privacy (Armstrong, 2002; Myles et al., 2003). Typical applications of LBS are found in areas such as navigation services (Winter et al., 2001; Chincholle et al., 2002; Winter, 2002; Choi & Tekinay, 2003; Smith et al., 2004) and tourist information systems (Zipf, 2002a, 2002b; Berger et al., 2003; Hinze & Voisard, 2003). In the area of emergency response, Erharuyi & Fairbairn (2003) present a conceptual framework for mobile spatial data handling in an oil spill management case study.

To a certain degree, LBS support decision-making by answering spatial queries. For example, the shortest route from the user's current location to a target location (e.g. restaurant) may be suggested. In addition, spatial queries can be combined with attribute queries, e.g. to further specify properties of the target location (e.g. Greek restaurant). However, multi-criteria methods in GIS go beyond querying by enabling users to evaluate and rank decision alternatives based on user preferences and the combination of multiple criteria.

Raubal & Rinner (2004) introduce a mobile hotel finder application that uses MCDA principles. Users specify decision-relevant attributes to be used as evaluation criteria. Next, users identify good, fair, and poor criterion scores or ranges to allow for comparison of standardized criterion scores. Finally, users define the relative importance of criteria by assigning weights. The weighted criterion scores are then combined based on a decision rule, resulting in an evaluation score for each decision alternative.

Rinner & Raubal (2004) extend the hotel finder application using the OWA decision rule that allows users to specify a personal decision strategy as part of their decision-related preferences. OWA defines a continuum from optimistic to pessimistic decision strategies in a mathematical sense and uses a second set of weights to emphasize high or low standardized criterion scores. For example, with a pessimistic strategy users would focus on the lower scores of each decision alternative (representing poor performance), while with the optimistic strategy, users would focus on the higher scores (representing good performance).

In a pub finder variant developed by Rinner et al. (2005), alternative user interface designs for location-based MCDA were developed to simplify mobile decision-making processes. For example, the manual standardization of decision criteria from the hotel finder application was replaced by automatic criterion standardization, and the continuous sliders used for criterion weighting in the hotel finder were replaced by a limited number of qualitative weight labels ("really want" to "really don't want").

16.4 Scenario of mobile decision-making in emergency response

Emergency management involves four phases: mitigation (prevention), preparedness, response, and recovery. GIS and related methods and tools, including positioning technology (GPS) and remote sensing imagery, are being used to various extents in those phases (Cutter, 2003). For example, spatial analysis methods can help with hazard identification and risk assessment in the mitigation phase. During the response to an emergency event, GIS is often used for data integration and mapping to support rescue and recovery operations (ESRI, 2002). Similar to desktop GIS, the use of mobile GIS in emergency response has also been limited to map creation to support field operations. In this chapter, an extension of mobile GIS with explicit decision support functionality, namely MCDA, is proposed and illustrated with an emergency response scenario.

Cai et al.'s (2004) hypothetical scenario of "geo-collaborative crisis management" is used to demonstrate this functionality. After a hurricane hit the coast of Florida, a first responder team uses mobile map-based communication with the emergency operations centre to find appropriate shelter for hurricane victims. The first responder team finds a group of elderly people that need to be evacuated from the flooded region to a shelter that provides certain services and has enough capacity. In Cai et al.'s scenario, the emergency operations centre compiles a map with shelter and background information that is shared with the first responders. The first responders request information on the capacity and caregiver staffing of an ad-hoc selected shelter and then decide to use that one as it "looks practical". This decision is based on the first responders' geographic intuition and facilitated by viewing the collaborative map.

As an extension of this scenario, we will assume that the emergency operations centre adds a shelter layer to the base map with shelter information as in the scenario by Cai et al. (e.g. current capacity, staffing). Further, the emergency operations centre sends a default decision strategy to the first responders' mobile GIS that consists of default importance weights for decision criteria such as the travel distance to shelters and the shelter attributes. Other settings for an MCDA method such as the decision risk could also be transmitted (e.g. low risk in the case of elderly hurricane victims). The first responders activate the MCDA process with a click and receive a suggested target shelter. On the map, several other shelters appear to be much closer to the first responder team's location. The first responders vary the decision strategy, for example by increasing the decision risk slightly above the suggested default. Soon, one of the nearby shelters becomes the top-ranked option and the first responder team takes measures to guide the victims to that shelter.

While this extension of Cai et al.'s scenario is not necessarily collaborative, it requires a server component that provides updated datasets and, possibly, organizational decision-making defaults. The scenario extension provides for more analytical structure in the location decision to be made by the first responder team than Cai et al.'s scenario.

16.5 Architecture of a map-based mobile decision support system

A mobile decision support system (MDSS) could be implemented as a standalone application on a handheld computing device. A snapshot of a geographic dataset together with decision models and a user interface would be installed on the handheld in a similar way to the setup of a car navigation system from a CD-ROM.

However, a few factors suggest conceiving of an MDSS as a client/server rather than a standalone application. Geographic data forms the basis of decision-making and needs to be as up-to-date as possible. For example, in the above scenario, capacities and staffing of shelters may change dramatically during a disaster. Therefore, it seems advisable to locate the geographic database component of the MDSS on a server under control of an emergency management centre as sketched in Fig. 16.2. The dashed database on the client side in Fig. 16.2 represents a local copy of data that is regularly updated and possibly extended with data that was collected by the current user of the handheld during the emergency.

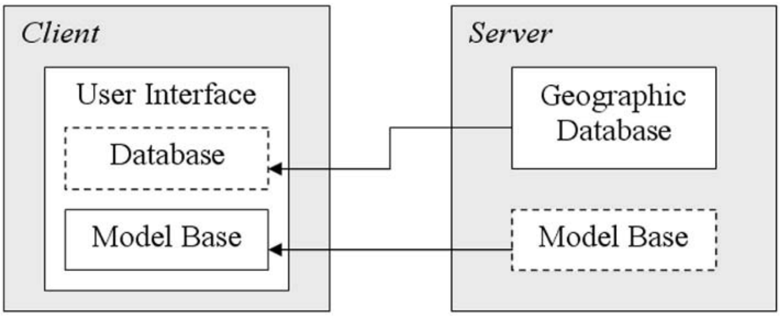


Fig. 16.2. Architecture schema for mobile decision support systems showing distribution of SDSS components over client and server

The multi-criteria methods contained in a model base of an MDSS can be mathematically as simple as a weighted averaging. This would suggest locating them on the client side as shown in Fig. 16.2. However, the server may play a role with respect to the model base when it comes to more complex simulation models that may be updated more frequently than MCDA methods, and for the storage and sharing of model parameters. On the one hand, personalized multi-criteria modelling as suggested by Rinner & Raubal (2004) requires storage of parameters such as importance weights and decision strategies. On the other hand, such parameter sets could be exchanged between users, or default settings could be offered by institutions as outlined in the scenario above. For example, an emergency operations centre may require a risk-averse (conservative) multi-criteria decision strategy when dealing with elderly people.

The user interface (UI) of an MDSS is naturally located on the mobile client. To provide access to the other two MDSS components, the UI needs a mapping component to display geographic data, and a dialog component to elicit the user's input of MCDA parameters such as the selection of decision criteria and the setting of criterion weights. Rinner et al. (2005) discuss UI elements of a common mobile GIS package that can be useful for representing the MCDA method on the mobile device screen. These include buttons, list boxes, combo boxes, sliders, and checkboxes. Rinner et al. have further examined options to simplify the UI by streamlining the MCDA process. For example, the choice of decision criteria could be automated and thus removed from the UI, and the standardization of criterion scores could be simplified to three levels chosen from a combo box rather than arbitrary level determined using sliders.

Fig. 16.3 shows the user interface of HotelFinder (Rinner & Raubal 2004), which consists of a series of tabs representing some of the steps in the MCDA workflow (Fig. 16.1). By contrast, Fig. 16.4 illustrates the simplification of the decision-maker's preference input in PubFinder (Rinner et al., 2005) with just one tab and qualitative labels for importance weighting. A design for the user interface of an MDSS of emergency response is presented in the following section.

Another component that has been identified as characteristic for SDSS is a report generator (Densham, 1991). However, we can assume that decisions made while moving are not of a scope for which a report is of importance, and therefore, the report generator does not appear to be a necessary component of an MDSS. In contrast, maps can serve as intuitive reporting tools in SDSS. Maps have been shown to aid individual and group decision-making, in particular in the reporting and presentation phases (Jankowski & Nyerges, 2001). In addition, interactive maps can be used to support data exploration (DiBiase, 1990; MacEachren, 1994; Andrienko & Andrienko, 1999) and spatial decision-making (Jankowski et al., 2001; Rinner & Malczewski, 2002; Malczewski & Rinner, 2005; Rinner & Taranu, 2006). Therefore, the user interface design for a mobile SDSS described in the following will include interactive mobile maps for the presentation and review of MCDA results.

Fig. 16.3. Multi-tab user interface of HotelFinder (Rinner & Raubal 2004)

Fig. 16.4. Simplified preference input in PubFinder (Rinner et al. 2005)

16.6 User interface design for a mobile decision support system for emergency response

Through the ArcPad Studio application, ESRI's mobile GIS ArcPad can be used as a generator for an MDSS. Extensions to ArcPad are provided through so-called "applets" (not to be confused with Java applets) which define new toolbars and forms. However, the available user interface elements are quite limited and the UI design shown in Fig. 16.5 to 16.7 and described in the following could not fully be implemented in ArcPad.

The UI design for this MDSS is centred on a map of the area surrounding the present position of the emergency response crew. As outlined in the introductory scenario, a first responder team is looking for the best shelter to evacuate a group of elderly people from a flooded area after a hurricane hit the coast of Florida. The emergency operations centre uploads a general reference map with facility locations and a multi-criteria decision support tool – Shelter Choice – to the first responders' mobile device.

The toolbar of Shelter Choice (top of mobile display in Fig. 16.5) provides basic mapping functions such as zoom to full extent, zoom in, and pan, as well as identify and layer management functions. A custom button enables the user to start the MCDA process.

The MCDA tool shown at the bottom of the map in Fig. 16.6 lets the user select criteria for determining the best shelter location from among the attributes of the facilities layer. Criteria shown in this make-up implementation include travel distance, capacity, staffing, and safety from being affected by the flood. The proposed design keeps the focus of the application on the mobile map by limiting the MCDA tools to an absolute minimum: the selection among available criteria and the weighting of the selected criterion. This approach requires the user to select criteria one after the other as long as he or she wants to modify their importance weights. In the example of Fig. 16.6, all four criteria start at a default weight of 25% that may have been determined by the emergency operations centre as an organization-wide default for this particular decision problem and this set of criteria. The fact that the map stays visible during the MCDA process allows the user to calibrate the MCDA results with his or her perception of the current situation as represented on the map.

Fig. 16.7 illustrates an option for the display of MCDA results. Locations from the facilities layer that are not in the top-ranked decision alternatives have been removed from the display in order to allow the user to focus on the best shelter locations. In the example, the three best shelters under the current, user-determined MCDA weighting scheme are shown by the school symbol and highlighted with a circle. The 'identify' tool is used to retrieve information about the top-ranked shelter. This information is reported with a label in the style of a tool tip.

Fig. 16.5. User interface design for a mobile decision support system for emergency response – map of possible shelters (schools, hospitals). Source of map data: Florida Geographic Data Library, http://www.fgdl.org/.

Fig. 16.6. User interface design for a mobile decision support system for emergency response – multi-criteria weighting. Source of map data: Florida Geographic Data Library, http://www.fgdl.org/.

Fig. 16.7. User interface design for a mobile decision support system for emergency response – display of results. Source of map data: Florida Geographic Data Library, http://www.fgdl.org/.

16.7 Conclusions and outlook

This chapter describes multi-criteria decision analysis as a useful addition to mobile decision support tools in an emergency response scenario. A client / server architecture for a mobile spatial decision support system was outlined and a map-centred user interface developed on the basis of the ArcPad mobile GIS. The ArcPad platform proved not to be flexible enough so that the MCDA interface had to be made up.

While Rinner & Raubal (2004) used a multi-page, modal form to gather user input for a large number of decision-making parameters, Rinner et al. (2005) suggested reducing user input by simplifying the MCDA method and criterion scales. Here, an even more efficient version of multi-criteria tools is proposed that is limited to setting the importance weight of one decision criterion at a time. This approach enables focussing on the situation map, thus it supports the first responder team in an emergency with calibrating the mobile tool with the current situation and their implicit spatial knowledge. Following a suggestion of Rinner et al. (2005), MCDA results are now also displayed in text form as labels on top of map symbols.

Through the application in emergency response, the proposed concept could demonstrate the utility of mobile mapping to support locational decision-making. According to the principles of geographic visualization, the MCDA result map ideally should be updated at every user action (i.e. changing the importance weights). In this way, the mobile map enables the decision-makers to quickly explore their choice options and thus becomes an efficient location-based decision support tool.

User tests are a top priority for further research on mobile decision support. The usability and utility of explicit spatial decision support tools such as MCDA needs to be demonstrated in realistic, or near-realistic, case studies. A second avenue for future development is the combination of multi-criteria methods to select a destination (such as a shelter) with navigation support methods that assist in reaching the selected destination. Of particular interest here is the combination with open standards such as the Open Geospatial Consortium's OpenGIS Location Service (OpenLS) specification.

Acknowledgements

Nampreet Kohli provided valuable support in generating the map for the user interface design. Amrita Hari helped to improve the manuscript. Partial funding by the Natural Sciences and Engineering Research Council of Canada (NSERC) is gratefully acknowledged.

References

Adams, P.M., Ashwell, G.W.B., and Baxter, R. (2003): Location-Based Services - An Overview of the Standards. *BT Technology Journal* 21(1): 34-43.

Ahn, Y.S., Park, S.Y., Yoo, S.B., and Bae, H.Y. (2004): Extension of Geography Markup Language (GML) for mobile and location-based applications. *Computational Science and its Applications - ICCSA 2004,* Lecture Notes in Computer Science 3044: 1079-1088.

Andrienko, G.L., and Andrienko, N.V. (1999): Interactive maps for visual data exploration. *International Journal of Geographical Information Science* 13(4): 355-374.

Armstrong, M.P. (2002): Geographic information technologies and their potentially erosive effects on personal privacy. *Studies in the Social Sciences* 27(1): 19-28.

Barnes, S.J. (2003): Developments in the M-commerce value chain: Adding value with location-based services. *Geography* 88(4): 277-288.

Beinat, E. (2001): Location-based Services - Market and Business Drivers. *GeoInformatics* 4(3): 6-9.

Benson, J. (2001): LBS technology delivers information where and when it's needed. *Business Geographics* 9(2): 20-22.

Berger, S., Lehmann, H., and Lehner, F. (2003): Location-Based Services in the Tourist Industry. *Information Technology & Tourism* 5(4): 243-256.

Cai, G., Bolelli, L., MacEachren, A.M., Sharma, R., Fuhrmann, S., and McNeese, M. (2004): GeoCollaborative Crisis Management: Using Maps to Mediate EOC–Mobile Team Collaboration. Demo Abstract, Proceedings of the 5th Annual NSF Digital Government Conference, Los Angeles, CA, May 23-26, 2004.

Chincholle, D., Goldstein, M., Nyberg, M., Eriksson, M. (2002): Lost or found? A usability evaluation of a mobile navigation and location-based service. *Human Computer Interaction with Mobile Devices,* Lecture Notes in Computer Science 2411: 211-224.

Choi W-J., and Tekinay, S. (2003): Location-Based Service Provisioning for Next Generation Wireless Networks. *International Journal of Wireless Information Networks* 10(3): 127-139.

Cutter, S.L. (2003): GI Science, Disasters, and Emergency Management. *Transactions in GIS* 7(4): 439-445.

Densham, P. (1991): Spatial Decision Support Systems. In D.J. Maguire, M.F. Goodchild, and D.W. Rhind (eds) *Geographical Information Systems: Principles and Applications,* Vol.1, pp. 403-412, London: Longman.

DiBiase, D. (1990): Visualization in the Earth Sciences. Earth and Mineral Sciences, Bulletin of the College of Earth and Mineral Sciences, Pennsylvania State University, 59(2): 13-18.

Erharuyi, N., and Fairbairn, D. (2003): Mobile geographic information handing technologies to support disaster management. *Geography* 88(4): 312-319.

ESRI (2002) New York City--Creating a Disaster Management GIS on the Fly. ArcNews Winter 2001/2002 (dated January 21, 2002). Available at http://www.esri.com/news/arcnews/winter0102articles/nyc-creating.html (accessed February 18, 2007)

Gartner, G., Uhlirz, S., Pammer, A., and Radoczky, V. (2003): Kartographische Beiträge zur Entwicklung von LBS. *Kartographische Nachrichten* 1/2003: 16-22.

Heywood, I., Oliver, J., and Tomlinson, S. (1995): Building an exploratory multi-criteria modelling environment for spatial decision support. In: P. Fisher (ed), *Innovations in GIS 2,* Bristol: Taylor & Francis, pp 127–136.

Hinze, A., and Voisard, A. (2003): Location- and time-based information delivery in tourism. Advances in Spatial and Temporal Databases, Lecture Notes in Computer Science 2750: 489-507.

Hjelm, J. (2002): Creating Location Services for the Wireless Web, John Wiley, New York.

Jankowski, P., Andrienko, N., and Andrienko, G. (2001): Map-centered exploratory approach to multiple criteria spatial decision making. *International Journal of Geographical Information Science* 15(2): 101-127

Jankowski, P., and Nyerges, T. (2001): GIS-Supported collaborative decision making: results of an experiment. *Annals of the Association of American Geographers* 91(1): 48-70

Jiang, H., and Eastman, J.R. (2000): Application of fuzzy measures in multi-criteria evaluation in GIS. *International Journal of Geographical Information Systems* 14(2):173-184

Keenan, P. (1997): Using a GIS as a DSS Generator, University College Dublin. Available online at http://mis.ucd.ie/staff/pkeenan/gis_as_a_dss.html (last accessed: 30 April 2006)

MacEachren, A.M., Ganter, J.H. (1990): A Pattern Identification Approach to Cartographic Visualization. *Cartographica* 27(2): 64-81.

MacEachren, A.M. (1994): Visualization in modern cartography: setting the agenda. In A.M. MacEachren and D.R.F. Taylor (eds) *Visualization in Modern Cartography.* Oxford: Pergamon Press, pp. 1-12.

Malczewski, J. (1999): *GIS and Multicriteria Decision Analysis.* New York: John Wiley & Sons.

Malczewski, J., and Rinner, C. (2005): Exploring Multicriteria Decision Strategies in GIS with Linguistic Quantifiers: A Case Study of Residential Quality Evaluation. *Journal of Geographical Systems* 7(2): 249-268.

Meng, L. (2005): Ego centres of mobile users and egocentric map design. In L. Meng, A. Zipf, and T. Reichenbacher (eds.) *Map-based Mobile Services. Theories, Methods and Implementations.* Berlin: Springer

Miller, H.J. (2003): What about people in geographic information science? *Computers, Environment and Urban Systems* 27(5): 447-453.

Mountain D., and Raper, J. (2001): Positioning techniques for location-based services (LBS): characteristics and limitations of proposed solutions. Aslib Proceedings: New Information Perspectives 53(10): 404-412.

Myles, G., Friday, A., and Davies, N. (2003): Preserving Privacy in Environments with Location-Based Applications. *IEEE Pervasive Computing* 2(1): 56-64.

Peng, Z.-R., and Tsou, M.-H. (2003): *Internet GIS: Distributed Geographic Information Services for the Internet and Wireless Networks.* Hoboken, NJ: John Wiley.

Pundt, H., and Brinkkötter-Runde, K. (2000): Visualization of spatial data for field based GIS. *Computers & Geosciences* 26(1): 51-56.

Raubal, M., and Rinner, C. (2004): Multi-Criteria Decision Analysis for Location Based Services. In S.A. Brandt (ed.) Proceedings of the 12th International Conference on Geoinformatics, pp. 47-53.

Rinner, C., and Malczewski, J. (2002): Web-enabled spatial decision analysis using Ordered Weighted Averaging (OWA). *Journal of Geographical Systems* 4(4): 385-403.

Rinner, C. (2003): Web-based Spatial Decision Support: Status and Research Directions. *Journal of Geographic Information and Decision Analysis* 7(1): 14-31.

Rinner, C., and Raubal, M. (2004): Personalized Multi-Criteria Decision Strategies in Location-based Decision Support. *Journal of Geographic Information Sciences* 10(2): 149-156.

Rinner, C., and Malczewski, J. (2005): Exploring multicriteria decision strategies in GIS with linguistic quantifiers: A case study of residential quality evaluation. *Journal of Geographical Systems* 7: 249–268.

Rinner, C., Raubal, M., and Spigel, B. (2005): User Interface Design for Location-Based Decision Services. In Proceedings of the 13th International Conference on Geoinformatics, 17-19 August 2005, Toronto, Canada.

Rinner, C., and Taranu, J. (2006): Map-Based Exploratory Evaluation of Non-Medical Determinants of Population Health. *Transactions in GIS* 10(4): 633-649.

Rinner, C. (in press): A Geographic Visualization Approach to Multi-Criteria Evaluation of Urban Quality of Life. *International Journal of Geographical Information Science*

Sarjakowski, L.T., and Nivala, A.-M. (2005): Adaptation to Context – A Way to Improve the Usability of Mobile Maps. In L. Meng, A. Zipf, and T. Reichenbacher (eds.) *Map-based Mobile Services. Theories, Methods and Implementations.* Berlin: Springer

Simon, H.A. (1977): *The New Science of Management Decision,* Upper Saddle River, NJ: Prentice Hall, 3rd edition

Smith, J., Mackaness, W., Kealy, A., and Williamson, I. (2004): Spatial Data Infrastructure Requirements for Mobile Location Based Journey Planning. *Transactions in GIS* 8(1): 23-44.
Spinney, J.E. (2003): Mobile positioning and LBS applications. *Geography* 88(4): 256-265.
Winter, S. (2002): Modeling Costs of Turns in Route Planning. GeoInformatica 6(4): 345-361.
Winter, S., Pontikakis, E., and Raubal, M. (2001): LBS for Real-Time Navigation - A Scenario. *GeoInformatics* 4(4): 6-9.
Yager, R.R. (1988): On ordered weighted averaging aggregation operators in multi-criteria decision making. *IEEE Transactions on Systems, Man, and Cybernetics* 18(1):183-190.
Zipf, A. (2002a): Adaptive context-aware mobility support for tourists. In Trends & Controversies: Intelligent Systems for Tourism. *IEEE Intelligent Systems* 17(6): 57-59.
Zipf, A. (2002b): User-Adaptive Maps for Location-Based Services (LBS) for Tourism. In K. Wöber, A. Frew, M. Hitz (eds.) Proceedings of the 9th International Conference for Information and Communication Technologies in Tourism, Innsbruck, Austria. Berlin: Springer. Heidelberg.
Zipf, A. (2003): Forschungsfragen zur benutzer- und kontextangepassten Kartengenerierung für mobile Systeme. *Kartographische Nachrichten* 1/2003: 6-11.