Personalized Mobile City Transport Advisory System

Gytis Tumas and
Francesco Ricci

Free University of Bozen-Bolzano, Italy
gytis.tumas@stud-inf.unibz.it, fricci@unibz.it

Abstract

Mobile devices are becoming an inseparable part of our lives and personalized location-based mobile services are gaining more and more popularity. The scope of this paper is to illustrate the design choices, the implementation, and the testing, of a personalized mobile city transport advisory system (PECITAS), built for the citizens and city guests of Bolzano, Italy. Using PECITAS the user can obtain, directly on his mobile phone, recommendations for personalised paths between two arbitrary points in the city. The paths are illustrated by listing the various connections that the user must take to reach the destination using public transport means and walking. The recommendations are selected in a personalized way, using a knowledge-based recommendation technology, and for each user the suggestions are computed according to their travel-related preferences. The concepts of travel and user profiles are introduced in order to rank different routes in the city and provide the top ranked to a user. PECITAS has been evaluated with respect to its suitability and efficiency in solving the routing problem showing that it can become a useful tool for city visitors and citizens.

Keywords: travel recommendations; city transport; tourism; location-based services.

1   Introduction

Optimal route computation in road networks is a common function of any GIS (Geographic Information System), i.e., an information system for capturing, storing, analyzing, managing and presenting data which are spatially referenced (linked to location) (Longley, 2001). Indeed the optimal route problem, i.e., typically the shortest route between two points, has been explored deeply, and there are plenty of software products, which are available as Web applications or mobile services, offering assistance in choosing the best route between two arbitrary points in a road network (Medhi and Ramasamy, 2007). One of the most popular examples of such service is offered by ViaMichelin (www.viamichelin.com). The main function of this system is finding the best route for a single transport mean (e.g., car or foot) between two arbitrary points on the map. By default the system focuses on road safety and comfort while offering a good compromise between time and distance. Additionally the user can specify his preferences by selecting the recommendation of the quickest, the shortest or the most economical path. The ViaMichelin functionalities are also offered nowadays by portable GPS based navigation systems, where the user is also supported in following the route by voice indications and annotated map interfaces. However one of the most important requirements for personalized tourist guides is providing transport information (Beer et al, 2007; Schmidt-Belz et al., 2003; Stroobants, 2006). With that respect the recommendation of the optimal route in a

network, where the user is allowed to use *several* transport means, e.g. busses, walking, car, etc., is a relatively new problem. Such a network where a user can swap the transport means is called *transit network* and it can be defined as a network that bridges at least two other types of networks. According to Ruihong (2007) there is still a lack of effective and efficient schedule-based path finding algorithms for transit networks.

Another popular line of research and development involves the concept of mobile tourist guides, i.e. applications running on mobile devices (PDA, smart phones) aimed at selecting Points of Interests (POIs) and guiding the user through a visit to them. Several mobile tourist guides have already been developed, however they do not focus on routing and transport advisory. For instance, MobiDENK (Krosche et al., 2004) – German acronym for *Mobile Monuments* – provides tourists with information about monuments. It presents to tourists a map of their environment, which is dynamically loaded from a GIS server and provides tourism related content stored in an internal database. The mobile client of the TourGuide system (Haid et al., 2007) provides the user with multimedia content related to the specific places that are situated around his route. Despite the variety of information the user is given about the route, the system cannot recommend a route itself according to the preferences of the user. *m-ToGuide* (Kamar, 2003) has been developed to promote the use of 2.5/3G cellular networks with Location-Based Services (LBS) and offers information about POIs. XML technology is employed to integrate content from different providers. *mobil-i* (http://j2mobile.blogspot.com/ 2006/12/mobil-i.html) has been developed in collaboration with the Public Transports of Geneva (TPG), and uses a mobile phone, connected to a Bluetooth GPS receiver, to display a map of the closest bus stops, as well as time tables and information about the next departures and how to reach the chosen bus stop by foot. However despite the nice interface the travel advice is limited to simply showing the information about the closest bus stops, i.e. there is no option for asking for recommendations about how to move between two arbitrary points using the public means. For example if a user knows her travel destination she cannot provide this information to the system and obtain in response a travel proposal. Conversely, the user must browse through the available bus routes and the reachable points in the city, thus no optimal routing function is offered.

Another interesting mobile service is Atac Mobile that was developed for travellers in Rome, Italy. The system provides different information on timetables of public transport (busses, metro), traffic situation, accidents, and it also offers a route finding function ("Calcolo del Percorso"). The optimal route is calculated given two arbitrary end points as input to the system (street addresses). The output is a detailed textual description of the trip, involving the walking distances, the bus lines to be taken, their arrival frequency, the places where to change the transport mean as well as the estimated travel time. Although this system does compute the shortest path in a transit based network (in this case the network of the public transport in Rome), the user involvement in selecting their preferences is very limited (she is allowed to select just the start and the destination points of the trip) hence the route is not personalized. For instance, if the user is a visitor, currently located at the train station, who intends to go to the Coliseum, having plenty of time to reach the destination, the system will

suggest the same route that will be given to a citizen that is moving to the destination for working purpose (i.e. using the underground), whereas a good combination of underground and walking would better serve the Rome visitor.

As this example shows, in real life situations users are not always interested in the quickest way to travel from point to point and other aspects could be more important when deciding on a specific trip in a city. For example, the price of the ticket or the fare (e.g. for a taxi) of the transport mean might influence the user's decisions. Moreover especially the visitors of a city are rarely interested in reaching a specific place in the city as soon as possible, and a longer route leading through the most famous POIs of the city might be much more convenient for the user, even if, for instance, she has to walk a bit more and the trip may take longer. In this paper, we are arguing that taking into account more diverse user's preferences and needs, a personalized solution may provide to the user a better trip recommendation.

This personalization issue can be addressed by a Recommender System. Recommender systems are intelligent E-commerce applications that assist users in their information-seeking tasks by offering personalized product recommendations during an interaction session (Adomavicius and Tuzhilin, 2005; Fesenmaier et al., 2006). However the majority of mobile recommender systems targeted for tourists, as we mentioned above, address just the problem of POIs selection. None of them considers the routing and the transport mean as a factor to be considered, and therefore they do not manage the routing itself as the recommendable item.

To demonstrate the feasibility and usefulness of such a function, a personalized mobile trip advisory system, called PECITAS (Personalised City Transport Advisory System), has been implemented for the city of Bolzano, Italy. It provides to the citizens and city guests of Bolzano user recommendations for the best route between two arbitrary points in the city, using several possible transport means, such as bus, taxi and walking. In this paper we present the design choices, the implementation, and the testing of this personalized mobile city transport advisory system. Specifically, Section 2 provides a high level description of the system. Section 3 focus on the personalization issues describing how the user and travel model is represented and how travels can be ranked. Section 4 illustrates the system evaluation and finally Section 5 summarizes the results of this work and indicates future planned developments.

2   System Description

The target users of PECITAS are mainly the city guests and the citizens who usually need a recommendation when on the move, i.e. without having access to a PC or the city transport schedules. The system consists of several components each responsible for different tasks (see Figure 1). The user interacts with the system running an application on his mobile phone, i.e. the PECITAS Java MIDlet. The user starts the interaction with PECITAS by choosing departure and destination points, as well as the departure time for the trip. The input of the user comprises also other preferences

(more details on these will be provided later) such as the walking time, maximal number of changes of the transport means, latest arrival time, and the sightseeing possibility.
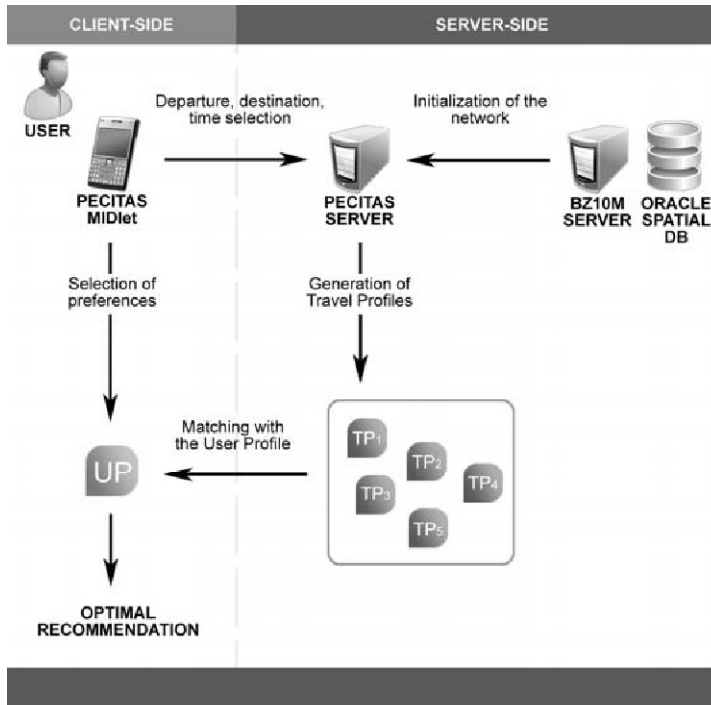

Fig. 1. System architecture

This information is sent to the PECITAS Server, where the transit network has been already initialized using the network data provided by another server, BZ10M server. This is an Oracle Spatial DB, and it is managing the data about the busses and roads of Bolzano. For each user request for a travel recommendation the implemented algorithm, running on the PECITAS server, computes some alternative routes, that we call travel profiles, since they can satisfy users with different travel preferences, and sends them back to the PECITAS client MIDlet. A travel profile is simply a description of a route, represented as a vector of travel features. The features describe the most important general characteristics of the route (e.g. length, number of POIs touched), that can be used to select the most appropriate travel for a given user. The travel profiles are then ranked (on PECITAS client) according to the specific travel preferences stated before (walking time, arrival time etc.). Finally the top ranked travel is recommended to the user, giving her detailed travel information. Figure 2 illustrates the GUI for entering the user preferences.
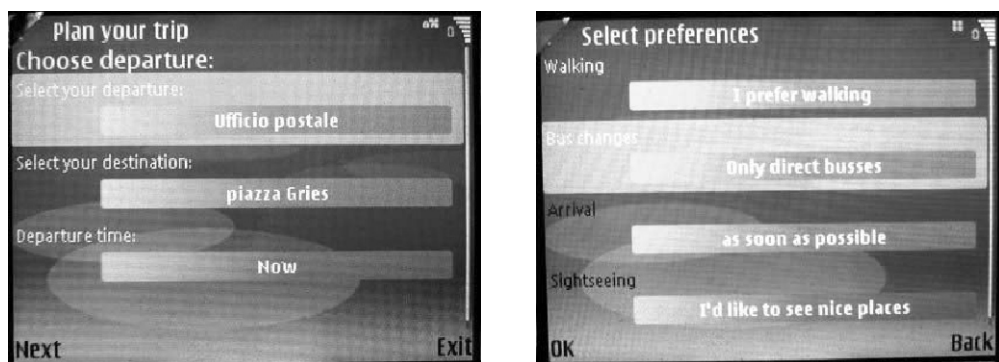
Fig. 2. PECITAS screenshots from the mobile device

3 Routing Personalization

As we mentioned above, the two main computations in PECITAS are performed by the routes generation and ranking algorithms. PECITAS routing algorithm is based on a single optimal route computation algorithm for transit networks (Ruihong, 2007). The main difference between PECITAS algorithm and that presented in Ruihong (2007), is that PECITAS generates *multiple routes* for a given departure and arrival points. PECITAS creates these multiple routes using some additional route constraints. The constraints were chosen in a way that the generated travel profiles are as diverse as possible, and at least one of them can match the preferences of the user collected in her user model. The following five types of travel profiles are generated by the PECITAS routing algorithm:

- TP_1 - the travel profile corresponding to the fastest route found by the algorithm
- TP_2 – the travel profile found by altering the departure time by +10 minutes
- TP_3 – the travel profile where the user is not allowed to take the busses in the city centre, therefore he is forced to walk through the central streets, increasing the number of touched POIs
- TP_4 – the travel profile where the user is not allowed to take any bus. The travel profile will consist only of walking
- TP_5 – the travel profile where the user is not allowed to use edges longer than 200 meters. The travel profile will clearly contain more walking, because the system is forced to choose shorter edges

The motivation for adopting this approach, i.e. to generate different travel profiles and then rank them, could seem inconvenient; why not immediately generating the best route, given the user preferences? There are two motivations for this choice. First, it was impossible to embed the user preferences as heuristics of the routing algorithm, because this is not a simple shortest path algorithm as it operates on transit networks. Secondly, users have rarely precise preferences at the beginning of the interaction, and preferences tend to be constructed during the interaction (Bettman et al., 1998). Hence, after the first recommendation is provided the user can decide to change her

travel preferences, such as the preferable arrival time, when seeing some possible travels. Having computed multiple travel profiles the system can easily update the recommendation, avoiding the calculation of a new travel profile.

The recommendation, i.e. the ranking of the generated routes, is based on the user preferences contained in the user model. To collect the user's preferences PECITAS poses explicit questions (optional) to the user (see Figure 2, for an example). Each time the user requests a recommendation for a travel she will be asked to specify her preferences. Four criteria with the following selection options are available when the user wants to get a recommendation:

- Walking:
  - I prefer walking
  - no longer than X minutes
  - as little as possible
  - I don't care
- Bus changes:
  - only direct busses
  - I don't care
- Arrival at the destination:
  - must before X o'clock
  - prefer before X o'clock
  - as soon as possible
  - I don't care
- Sightseeing:
  - I'd like to see nice views/objects
  - I don't care

These options are used to generate the user profile that is basically a vector of preference features. After different travels are generated at server-side, the resulting travel profiles, e.g. TP_1, TP_2, …, TP_5, are returned to the PECITAS client, where the final selection of the best travel profile is done. If there are two or more profiles having the same characteristics (e.g. TP_3 might be equal to TP_1 in a case when TP_1 consists only of walking) then fewer profiles will be ranked. The matching of the user model and the travel profiles, which follows a knowledge-based approach (Burke, 2007), is done using a set of predefined functions that map the values of the attributes of TP_n ($n=1$, …, 5) to a satisfaction score for the user (see later). To collect the user preferences and obtain the travel recommendation the human-computer interaction follows these steps:

- *Identify the start point.* This can be done by either choosing a place from a list of predefined places (such as the stations, hotels, squares, shops etc.), or by choosing an existing bus stop or simply by choosing the current location (using the GPS information).
- *Identify the destination point.* This, as illustrated before, can be chosen either from a list of predefined places, or by selecting a bus stop.

- *Select the departure time.* The default option is to select the current time of the user query, although the user has the possibility to plan the travel with a departure time different from the current one. We must note that choosing a different departure time might dramatically influence the recommendation results of the system, depending on the schedules of the busses and the locations of the departure and destination.
- *Select the travel preferences.* The travel preferences, described below, are used to construct the user model and personalize the recommendation.

The user model comprises six attributes $(u_1, u_2, …, u_6)$ that are determined by four general travel preferences: walking, bus changes, arrival at the destination, and sightseeing.

1. Walking preferences are used to set u_1 and u_2. u_1 is a nominal attribute taking one of the following four values:

- *w* - if the user has selected that she prefers walking,
- *mw* - if the user wants to walk no more than some specified amount of minutes,
- *lw* - if the user wants to walk as little as possible and
- *na* - if the user does not care about walking.

In case the user selected the *mw* option, u_2 is a numeric attribute representing the maximum number of minutes the user is willing to walk.

2. Bus changes preferences are used to set u_3. This is a nominal attribute with two possible values. The value *d* is assigned if the user prefers to use only direct busses. If the user does not care about this option, the *na* value is assigned to this attribute.

3. Arrival at the destination preferences are used to set u_4 and u_5. u_4 is a nominal attribute, describing the user's preferences with respect to the arrival time at the destination. The value *bt* (*pbt*) means that the user would like to be at the destination strictly (or possibly) before the specified time. The value *asap* stands for "as soon as possible". Finally, if the user does not care about the arrival time the value *na* is assigned.

In case the values *bt* or *pbt* were chosen for the u_4 attribute then a fifth numeric attribute u_5 represents the latest preferred arrival time.

4. Sightseeing preferences are used to set u_6. This is another nominal attribute, showing the user's interest in seeing the most interesting POIs in the city. The *y* value is assigned if the user has chosen that she would like to see nice views/objects, otherwise its value is *na*.

Thus, the user profile is composed of the six attributes $UP = (u_1, u_2, …, u_6)$. As we mentioned above, the recommendation algorithm first computes some routes connecting the departure and arrival points and then scores them by matching them to

the user model description. To match the user profile with a route, this is represented as a travel profile $TP_n = (t_1, t_2, t_3, t_4)$ consisting of four attributes. The attribute t_1 represents the walking time in minutes, the attribute t_2 is the number of busses used in the travel, t_3 stores the time of the day when the destination is reached, and finally t_4 represents the number of POIs that are touched by the travel. It should be noted that the walking speed is set to 5km/h that is an average human walking speed.

The overall satisfaction score for a travel profile TP_n is defined as follows:

$$S(TP_n, UP) = \frac{\sum_{i=1}^{4} f_i(TP_n, UP)}{\# of f_i \neq \emptyset}$$

where TP_n is a travel profile, UP is a user profile, and f_i are the functions mapping each one of the four general travel preferences (walking, bus changes, arrival at destination, sightseeing) to the appropriate satisfaction score of the user for the travel profile TP_n. Each of the four functions $f_1(TP_n; UP), \ldots, f_4(TP_n; UP)$ operates on the different travel preferences and behaves differently according to the user preferences.

A complete description of all the four f_i functions would require too much space. We illustrate one example to illustrate the general approach. So, if the user has said that she prefers walking, the function depicted in Figure 3 will determine the value of $f_1(TP_n; UP)$. Let us assume that the user prefers walking but not more than 90 minutes, in this case we also assume that he would be pretty satisfied walking at least 30 minutes. This is shown by the fact that in this case f_1 increases linearly in the range from 0 to 30 (minutes), and reaches the maximum of 1 at 30. Then, walking more than 90 minutes would linearly decrease the satisfaction, even for the user who likes walking. When the walking time reaches 180 minutes the satisfaction score will get the minimum value of 0. Therefore, the maximum satisfaction score will be given to the travel profile where the user has to walk between 30 and 90 minutes.
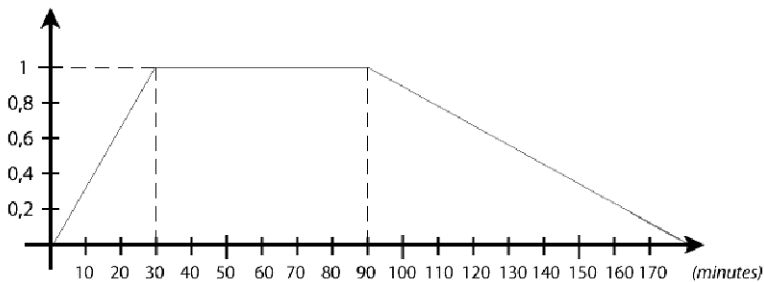

Fig. 3. Scoring function for the walking preference in case the user likes walking

Note that the function $f_1(TP_n; UP)$ considers the different user preferences and it would behave differently if the user has chosen that she does not like to walk or she has chosen any other different walking preference.

We have defined similar functions for all the four general travel preferences after a series of trial and test experiments. Obviously, the optimal definition of these

functions could be a matter of further studying, improvement, and learning, as the system is used by a population of users.

4 Evaluation

We tested PECITAS using the data of the transport network of Bolzano (Italy). This network has 2.815 nodes, 3.526 links, 17 bus lines and 15.382 bus departures. In order to test the speed of the recommendation process 50 different queries were run, each one with a random selection of the departure and destination points, and departure times. The average running time of the algorithm to compute the five different travel profiles TP_1, TP_2,…, TP_5 was 8,4 seconds. The test was performed on a PC with a 1,6 GHz CPU and 512 MB of RAM. We must observe that in some cases some of the generated travel profiles were identical. For instance, if the fastest path does not lead through the city centre, then the travel profile ignoring the possibility to use the bus in the city centre, is the same as the fastest path profile. Or, when the departure and destination points are close enough, the fastest route is by walking, hence TP_1 is equal to TP_4, i.e. the travel profile that ignores the busses.

We tested the usability and efficacy of the system with some students of the Computer Science Faculty of the University of Bolzano, without raising major usability issues. Here, to illustrate the usage of the system, we run an example usage scenario for a particular recommendation. Imagine that a couple of tourists are in the train arriving to Bolzano in 10 minutes, and the current time is 18:03. Using PECITAS on their mobile phone they query the system for the optimal way to reach Piazza Gries leaving from Bolzano train station, using the public city transport. They select the departure and destination points and specify the departure time to be in 15 minutes. With respect to their travel preferences they state that: they do not want to walk more than 30 minutes; they want to take only direct busses; they want to arrive at the destination as soon as possible; and they want to see nice places and objects.

After these preferences are sent to the server, PECITAS starts generating the possible travel profiles. The TP_1 profile suggests going to bus stop 3, waiting 6 minutes, at 18:25 getting on the bus number 10A, at 18:32 boarding off at Piazza Gries bus stop and walking additional 90 meters to the destination. The TP_2 (travel profile with the altered departure time by 10 minutes) suggests walking to the same bus stop 3, at 18:29 getting on the bus number 2, getting off in Piazza Walther at 18:30, getting on the bus number 10A at 18:46, at 18:52 getting off at Piazza Gries bus stop and walking to the destination. TP_3 (the travel profile that ignores the possibility to use the bus in the city center) suggests walking 1481m to the Vittoria bus stop, at 18:48 getting on the bus number 10A, at 18:52 getting off at Piazza Gries bus stop and walking to the destination. TP_4 (ignores all busses) suggests walking 2650m to the destination. TP_5 (ignores the streets longer than 200 meters) in this case is identical to TP_1, because on the fastest path there are no street segments longer than 200 meters.

After the matching between the retrieved travel profiles and the user profile was done, TP_3 travel profile was recommended to the user, because it has the highest satisfaction

score among all travel profiles. This is a logical recommendation, because TP_3 suggests passing by three interesting points and the arrival time to the destination is relatively early, therefore the selected profile provides the best trade-off among all the user preferences.

5   Conclusions and Future Work

In this paper we have presented PECITAS, a system that helps users to find a personalized path connecting two arbitrary points of a city using the city transport means and walking. While assisting the user in travelling in a city using public transport means has been already offered by other systems, most of these applications cannot suggest the best way to travel between two points taking into account the user preferences. In fact, a particular itinerary could, for instance, be shorter in time but could require the user to change line several times, or could be just a bit faster than walking. A user may prefer one of these solutions taking into account other preferences in addition to that related to time minimization. PECITAS is targeted to users of mobile devices and provides a location-based service recommending the optimal path, based on the specific user preferences. To implement PECITAS, we designed and implemented a fastest-path algorithm for transit networks, and a route ranking methodology based on a particular user model and a route model. In the experiments that we conducted, we used the full data set of the busses and roads of Bolzano. Because of the size of the database and the number of queries performed, we observed that the initialization of the algorithm would require 4-6 minutes, however this is done just once and after that the fastest path finding algorithm can be run by different users without any further delay. We have also shown that a solution of the user problem can be found on average in 8,4 seconds. And this includes the route computation, path ranking and client-server communication times. This is acceptable and makes the solution feasible in a real mobile scenario.

There are several aspects that would deserve some further work. First of all the system can be extended to support other transport means, such as cycling, taxi, etc., not restricting the user by only walking and bus means. The visual description of the result on a map would be a feature that allows an easier interaction with the system. In addition, the generation of a reasonable diverse and meaningful set of travel profiles still has to be explored better. The algorithm should be improved in order to retrieve more reasonable travel profiles, which are diverse enough from each other, so that the recommendation would be more precise and could better satisfy the user needs. Finally we plan to conduct a user study, to test the usability of the proposed solution with a real set of citizens and visitors of Bolzano.

References

Adomavicius, G., and Tuzhilin, A. (2005). Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Engineering, 17*(6), 734-749.

Beer, T., Fuchs, M., Höpken, W., Werthner, H., and Rasinger, J. (2007). CAIPS: A Context-Aware Information Push Service in Tourism. *Information and Communication Technologies in Tourism 2007*. Springer, 129-140.

Bettman, J. R., Luce, M. F., and Payne, J. W.. (1998) Constructive consumer choice processes. *Journal of Consumer Research: An Interdisciplinary Quarterly*, 25(3):187-217.

Burke, R. (2007). Hybrid Web recommender systems. *The Adaptive Web,* Springer-Verlag, 377-408.

Cormen, T. H. and Charles E. Leiserson and Ronald L. Rivest. (1992) *Introduction to Algorithms.* McGraw-Hill.

Fesenmaier, D. R.,Werthner, H., and Woeber, K. (2006). *Destination Recommendation Systems: Behavioural Foundations and Applications*. CABI Publishing.

Haid, E., Kiechle, G., Goll, N., & Soutschek, M. (2007). Evaluation of a Web-based and Mobile Ski Touring Application for GPS-enabled Smartphones, *Information and Communication Technologies in Tourism 2008,* 313-323.

Kamar, A. (2003). Mobile Tourist Guide (m-ToGuide). Deliverable 1.4, Final Report.

Kang, Y., Stasko, J., Luther, K., Ravi, A., & Xu, Y. (2007), RevisiTour: Enriching the Tourism Experience With User-Generated Content, *Information and Communication Technologies in Tourism 2008,* 59-69.

Krosche, J., Baldzer, J., Boll, S. (2004). MobiDENK – Mobile Multimedia in Monument Conservation. *IEEE MultiMedia*, 11(2), 72-77.

Longley, P., (2001). Geographic Information Systems and Science. Wiley.

Medhi, D. and Ramasamy, K. (2007). Network Routing: Algorithms, Protocols, and Architectures. Morgan Kaufmann.

Ruihong, H. (2007) A Schedule-based Pathfinding Algorithm for Transit Networks Using Pattern First Search. *Geoinformatica,* 11(2), 269-285.

Schmidt-Belz, B., Laamanen, H., Poslad, S., & Zipf, A. (2003). Location-based mobile tourist services-first user experiences. *Information and Communication Technologies in Tourism 2003.* Springer, 115-123.

Stroobants, R. (2006). Mobile Tourist Guides. *Katholieke Universiteit Leuven, Leuven.*