A Mobile 3D-GIS Hybrid Recommender System for Tourism

José M. Noguera, Manuel J. Barranco, Rafael J. Segura, Luís Martínez
A Mobile 3D-GIS Hybrid Recommender System for Tourism

José M. Noguera*, Manuel J. Barranco, Rafael J. Segura, Luís Martínez

Department of Computer Sciences, University of Jaén
Campus Las Lagunillas s/n, Jaén, 23071, Spain.

Abstract

The amount of touristic and travel information existing on the Internet is overwhelming. Recommender systems are typically used to filter irrelevant information and to provide personalized and relevant services to tourists. In this context, mobile devices are particularly useful because of their ubiquitous nature that turns them into an attractive platform for assisting on-the-move tourists to choose points of interest to visit according to their physical location. However, mobile devices also present several usability limitations that should be considered in order to provide information in a direct, intuitive way. In this paper, we present a novel mobile recommender system that brings together a hybrid recommendation engine and a mobile 3D GIS architecture. This system allows tourists to benefit from innovative features such as a 3D map-based interface and real-time location-sensitive recommendations. The details related to the design and implementation of the proposed solution are also presented, along with an empirical evaluation of user experience with the

*Corresponding author, Phone: +34 953 21 28 53, Fax: +34 953 212 420

Email addresses: jnoguera@ujaen.es (José M. Noguera), barranco@ujaen.es (Manuel J. Barranco), rsegura@ujaen.es (Rafael J. Segura), martin@ujaen.es (Luís Martínez)
mobile application.

Keywords:
Mobile computing, e-tourism, Recommender systems, Mobile map, 3D maps

1. Introduction

In the last decade, the growth of the Internet and its users has heavily modified several social behaviors in many fields. In tourism the web has originated the electronic tourism (e-tourism) [70, 71]. According to recent studies (e.g. European Travel Commission), the web is the primary source of information for people in Western countries for planning trips and obtaining information about travel destinations [31]. Despite this fact, most tourism web sites only offer basic booking functionality, and are unable to support customers in their searches. Clearly, complexity and quantity of e-tourism products might become a problem because users cannot look up entire databases in order to choose the most suitable products for them. Therefore, frustrated users will likely give up the web site without any purchase.

Personalization [22] has recently been recognized by experts like a critical factor of e-tourism companies to be successful. Different personalization techniques have emerged to deal with this problem. The most successful one has been the recommender systems (RS) [2, 55, 62], which lead users to suitable products by using different information filtering techniques that utilize knowledge about their personal preferences. The most popular approaches in e-tourism personalization are collaborative filtering and content based techniques [21, 56].
Recently, we proposed REJA [43, 59] a web-based restaurant georeferenced RS for the province of Jaén in southern Spain. REJA provides basic personalization services to those tourists that visit Jaén paying attention to the cold start problem [4, 69] and overcoming it by a hybrid technique based on a knowledge based approach. However, new challenges in the e-tourism industry require these services to be available without spatial-temporal limitations and to provide context-aware information [20].

These new e-tourism challenges have been tackled by using mobile computing (e.g. mobile phones, tablets, personal digital assistants, etc.) [21, 33, 57, 63]. Because such platforms may add *ubiquity* providing services to customers at anytime and anywhere; *connectivity* with wireless networks (e.g., GPRS and UMTS) at a relatively low cost [16]; *location-awareness* by means of Global Positioning Systems (GPS) and compasses that enable to take contextual knowledge (physical location, motion speed, time, etc.) of users into consideration [3, 21]; and *graphics capabilities* because recent high-end mobile devices include Graphics Processing Units (GPUs) and large display sizes, which open the door to the development of 3D user interfaces [5].

Context-aware recommender systems (CARS) are a special type of RS that takes users’ contextual knowledge into account when providing recommendations. Early CARS designed for mobile devices mostly rely on the user’s location to determine the appropriateness of POIs to be provided to the customers, see for example [1, 17, 32, 52, 64, 68]. However more advanced solutions are capable of taking advantage of customer’s previous choices information by means of content-based filtering [37, 61, 72] or exploiting knowledge about other customers with similar interests by means of collaborative
filtering [10, 29, 30, 34].

Apart from a recommender engine, the success of a mobile tourism support system depends on an intuitive and usable interface. So far, because of the limited size of mobile devices, user interaction is especially difficult and frustrating. All reported solutions portray recommended items within a textual interface or, at best, within an abstract 2D map [33]. However, both approaches either present severe usability problems or require a cognitive effort to understand the provided information.

Taking into account the previous features and drawbacks presented by current mobile e-tourism personalized systems, we present different extensions upon our system REJA in order to address different aspects related to mobile recommender systems and overcome interface drawbacks. First, we extend our system to include user’s location, such that, the recommendations are restricted to an influence area determined by the user’s location together with her preferences and generated by a hybrid recommender approach. Second, we provide an interface for georeferencing recommendations with a 3D view of the terrain with actual imagery (satellite/aerial) and landmarks that would be easier and faster to understand [49] by using a solution integrated into a mobile 3D GIS. This allows users to benefit from innovative features including 3D geovisualization, progressive transmission of terrain across wireless networks, geolocation and possibility to download location-sensitive recommendations to their mobile devices on demand. Third, this paper describes and discusses a complete client-server architecture which implements these ideas. Finally, the results of an evaluation survey about the system performance and usability are presented.
We must highlight that contrarily to previous solutions that investigated mobile recommender systems using a desktop-based system simulating a mobile device (e.g., [37, 72]), we have carried out our implementation and evaluation on real smartphones. Therefore, the proposed algorithms and strategies have been designed aiming at efficiency, simplicity and usability on small display sizes.

The rest of the paper is organized as follows. Section 2 provides a necessary background about recommender and geographic information systems. Section 3 presents our proposal of a georeferenced hybrid recommender system designed for mobile tourism. In order to validate this proposal, we have built a fully operational prototype. Section 4 introduces this prototype and outlines its design and architecture. Section 5 presents a survey about the usability and performance of the proposed system. Finally, Section 6 concludes the paper.

2. Preliminaries

This Section provides a brief review about recommender systems describing the most common models, paying then attention to the main features and problems of the systems used by our proposal. Following, due to the fact that our system makes heavy use of GIS and 3D mobile technologies, some general concepts are provided.

2.1. Recommender Systems

Recommender Systems assist people to find out suitable products in e-shops with large databases of items. These tools aim at providing customers
with useful recommendations according to their necessities and tastes. Typically, recommendations are inferred based on one or several of the following information sources [6]:

1. User’s preferences among alternative products.
2. User’s preferences about product attributes.
3. Other people’s preferences or choices.
5. Individual characteristics that may predict preferences.

Depending on the information source and the technique used to rank items, there are different types of recommender systems:

- **Demographic recommender systems** [36]: They categorize users into demographic groups based on personal attributes. Users receive recommendations according to the group in which they are classified.

- **Content-based recommender systems** [51]: This type of recommender systems computes recommendations according to the features of items that the user preferred in the past.

- **Collaborative filtering recommender systems** [24]: These systems use customer’s ratings to categorize users in groups according to their similarity. Recommendations are then inferred by taking into account ratings of users’ in the same group.

- **Knowledge based recommender systems** [14]: These systems infer recommendations based on user necessities and on knowledge about item
features. These systems typically compute their recommendations using case based reasoning processes, i.e., the user provides an example of an item similar to his/her interest and the system then infers a profile in order to find the best matching product in the search space.

- **Utility based recommender systems** [26]: They compute recommendations based on the calculation of the utility of each item according to user interests.

- **Hybrid recommender systems** [9, 15, 40]: The aforementioned systems suffer from limitations under certain circumstances. Hybrid recommenders try to avoid such limitations by combining two or more different approaches.

According to recent research, collaborative filtering (CF) [24] is the most successful recommendation technique up to date [42, 54]. There are three general tasks that a CF approach must fulfill in order to provide recommendations:

1. Analyzing and selecting data sets [28].
2. Grouping users according to their tastes and preferences.
3. Generating predictions for the target customer by using different aggregation methods [2, 27].

There are two main advantages that make CF one of the most popular and efficient recommendation technique:

(a) Ability to work, even with items whose contents are unknown or difficult to obtain.
(b) It can provide recommendations “outside the box”, i.e. recommended products can be quite different from products positively rated by the user. Other recommender systems, like content-based ones, never recommend products outside the box.

In spite of the general good performance of such systems, they present several weaknesses and limitations:

(a) *Grey sheep problem*: Grey sheep users that are in the frontier of two groups of users do not obtain good recommendations.

(b) *Historical data set*: The adaptivity of these systems because of historical data is an advantage but may be also a disadvantage when the historical data set is small.

(c) *Cold-start problem*: This problem is presented with both users and products. When a new user access to the system, it has not any information about him/her or it is scarce. Therefore, the system cannot compare him/her with the users of the database and cannot provide recommendations [44]. When a new item is added and no user has assessed it yet, it cannot be recommended [67].

To overcome such problems, and mainly the cold-start one, different hybridizing proposals have been presented [4, 15, 60]. In [43, 59] the recommender systems REJA presented a commutation hybridizing approach between a collaborative and a knowledge based systems to tackle cold start situations and then obtain easily recommendations with scarce information.
2.2. Context-aware Recommender Systems

Most classic existing recommender systems only take into account two type of entities: users and items, and do not consider any contextual information such as location, time, weather, companion, etc. Context-aware recommender systems, on the contrary, aim at providing users with relevant information according to their current physical contexts. CARS is an active research field. Recent surveys published by Ricci [57], Adomavicius et al. [3] and Felfernig et al. [21] provide a broad overview of this area.

Recently, several authors have studied the usefulness of exploiting contextual information in the recommendation field. In [3], Adomavicius et al. proved that contextual information is an important tool for increasing the quality of recommendations in certain settings. In an user study, Baltrunas et al. [8] found that users find more effective and satisfactory a CARS than a standard RS with the same user interface. Similarly, empirical experiments performed by Gorgoglione et al. [25] also demonstrated that CARS produce better results than standard RS in terms of customer purchasing behavior and trust in the provided recommendations.

The concept of context-awareness matches with the ubiquitous nature of mobile devices, and specifically, with the knowledge of the user’s physical location, orientation, speed, etc. that these devices provide. Moreover, as handheld devices feature very limited displays and input capabilities, RS can play an important role to reduce the information overload. Therefore, mobile tourism guides are the most usual application of CARS [20, 57]. A mobile guide supports travelers in planning, organizing and executing a journey [8]. As the user moves, the guide helps him to find relevant services and
information according to the location being visited. Nevertheless, CARS are also being actively used in additional fields, e.g., for suggesting music according the user mood or location [7, 12, 66] or for proposing stores [72] and mobile applications to purchase [11].

Early mobile guides focused on exploiting context information to determine the appropriateness of POIs, but lacked an actual recommendation engine. See for example Cyberguide [1], Guide [17], Crumpet [52] or Compass [64].

However, more recent systems also took into consideration users’ tastes and preferences. The aim of these systems consists of incorporating available contextual information into the modeling process of the recommendation. Typically, either content-based or collaborative filtering is used for this purpose. Content-based recommenders aim at taking advantage of customer’s previous choices. Recent proposals in this category include I’m feeling LoCo [61] and the proposals of Kuo et al. [37] and Yang et al. [72]. On the other hand, collaborative recommenders exploit knowledge about other customers with similar interests, see for example GeoWhiz [29], MyMytilene [34] or the proposals of Biuk-Aghai et al. [10] and Huang et al. [30]

According to Adomavicius and Tuzhilin [3], there exist three approaches for incorporating context information in RS:

- **Contextual pre-filtering**: Context information is used to select the most relevant items. Predictions are then generated by using any traditional recommendation model on the selected items.

- **Contextual post-filtering**: Contextual information is initially ignored, and a list of items is generated using any traditional recommendation
model. Context information is then used to filter out irrelevant items or to rerank them.

- **Contextual modeling**: Contextual information is used directly in the modeling process.

In our proposal, we will use both pre and post-filtering approaches.

### 2.3. Mobile 3D Geographic Information Systems

According to Demers [19], a geographic information system is a software system designed to input, store, edit, retrieve, analyze and output geographically referenced data, i.e., data tied explicitly to known points on the earth’s surface. GIS are valuable tools for tourism services because products offered in this commerce sector are usually places which can be located on a map, e.g., accommodation, restaurants, monuments, etc. Consequently, most advanced mobile tourist guides reported in the literature use 2D GIS technologies to represent spatial data [33]. These 2D map-based interfaces typically provide users with the ability to graphically visualize geographic entities and recommended items in an intuitive way, allowing them to assess how far items are from their current position and among them.

Unfortunately, 2D map-based interfaces also have some problems. Mobile devices with small screens have limitations on the size of the geographic area that they can display at one time [13]. Therefore, a global view of area around the user is only possible by reducing the scale of the map. But as a tradeoff, users may find this scaled-down representation confusing because of the loss of detail and the high amount of information simultaneously displayed on the screen. Eventually, 2D maps also require cognitive resources and topological
reasoning to read the mobile map and relate it to the environment that surrounds the user [49].

Recently, the popularity of 3D GIS solutions for desktop computers has rapidly increased due to the fact that 3D representations provide an immersive user experience and permit real-time fly-throughs, among other benefits [41, 49]. Note that although 2D maps are frequently rendered under a false perspective projection (e.g., on most car navigation systems), these maps are not true 3D representations because the three dimensional characteristics of the environment are missing.

In the mobile field, however, battery-based handheld devices have historically suffered from severe hardware constraints which left them outside of the 3D GIS market. Fortunately, the advent of mobile graphics processing units (GPUs) during the last 2-3 years has boosted the graphics capabilities of mobile devices [5, 16], opening the door to the development of mobile 3D maps based applications which were unbelievable just some years ago, see for example [47, 48].

Technically, 3D maps are commonly implemented by using digital elevation models (DEMs) [19, 41] and realism is further enhanced by adding photo textures, consisting of actual aerial/satellite imagery. However, the ever increasing size and resolution of terrain datasets widely exceeds the capabilities of any computer. Therefore, out-of-core solutions are widely used, which store the complete dataset in a remote server, from where the required parts of the terrain are progressively streamed to the client over a network, see [50] for further details.
3. A 3D-GIS Mobile Location-aware Recommender System

Here a proposal is introduced for a 3D context-aware mobile recommender system. This proposal could be applied to different e-tourism purposes, but in this paper we upgrade the REJA system [43, 59] from a web-based RS to a 3D map-based mobile CARS whose goals are:

- **Ubiquity**: Users may use the system wherever they like. The mobile platform and GIS are key factors to achieve this goal.

- **Location-awareness**: because of the ubiquitous nature of the system, it seems necessary and convenient that the recommendations provided are adapted to the user’s current location.

- **3D-interface**: the usability of mobile applications is crucial for its success. Here we propose a 3D solution with innovative features as 3D geovisualization, location, etc.

Originally, REJA was formed by a commutation hybridizing system with collaborative and knowledge filtering. Location information has been integrated following two different approaches [3]. The new schema is graphically showed in Figure 1, and it involves the following phases:

1. **Location-aware Hybrid Recommender System**: First, a contextual pre-filter is used to reduce the number of items considered for the recommendation according to the user’s location. These items are used by our RS to generate a top-N list of items potentially interesting for the user, see Section 3.1.
2. Distance based re-ordering: Then, a contextual post-filter re-orders the previous top-N list according to the physical distance from the user to each item, as described in Section 3.2.

In what follows, we present in further detail the performance of the these two phases of the proposed recommender system and eventually, it will be described the 3D user interface.

3.1. Location-aware Hybrid Recommender System

One of the main objectives in our mobile recommender system proposal is to add the capacity of adapting recommendations to the current user’s location. To do so, we propose the use of a location-aware pre-filtering module that defines the area considered by the system to provide recommendations, see Section 3.1.1. This module will be integrated in both collaborative and knowledge based processes to adapt their recommendations to the considered
area, see Sections 3.1.2 and 3.1.3, respectively.

Below we describe its performance and integration in both recommendation techniques.

3.1.1. Location-aware Module

When tourists are visiting a scenic place, they usually move around a limited area that will depend on their transportation (walking, car, bus, etc.). Our module will fix such an area by means of a parameter, so called $R_{outer}$, that establishes the limits where the items might be recommended.

We assume that our recommender system deals with a set of recommendable items, $A = \{a_1, ..., a_n\}$, geographically located. Therefore the parameter, $R_{outer}$, will be used to compute a subset, $A' \subseteq A$, that includes those items that could be suitable to be recommended to the user according to her location and ignoring the remaining ones because they are far away from user’s location (see Fig. 2).

The parameter, $R_{outer}$, will be defined by the user and can be modified whenever she likes.

3.1.2. Location-aware Collaborative Filtering

Collaborative recommender systems carry out three phases to compute their recommendations [27]:

---

*Figure 2: Area for recommendations: Items $a_4$ and $a_5$ are discarded because they are outside.*
1. **Grouping users**: the users registered in the system are grouped according to their preferences. Hence, those users with similar tastes belong to the same group.

2. **Generating predictions**: for a given user that requires recommendations it is computed a prediction for every item that she did not experience yet.

3. **Recommendations**: according to the previous predictions, the item or the top-N items with best prediction are recommended to the user.

However to include the process of location-awareness, the previous collaborative process is modified in order to keep recommending the best items but taking into account that such recommendations must be in $A'$, i.e., the distance from the user’s location to the item must be less than $R_{outer}$. To fulfill this condition the previous *Generating prediction* has been modified as:

2. **Location-aware generating predictions**: the predictions are computed only for those items in the limited area such that, $a_i \in A'$, and have not been experienced yet by the user. This process is graphically showed in Figure 3.
Due to the fact that the number of items in $A'$ might be scarce, occasionally the collaborative module cannot obtain recommendations. In such a situation our hybrid system commutes to the knowledge based system to provide recommendations to the user.

3.1.3. Location-aware Knowledge Based Filtering

Cold-start or scarce information situations make the collaborative module cannot provide recommendations to the user. The use of knowledge based recommender systems might be used to overcome such a limitation of collaborative systems [15, 50]. Knowledge based systems perform different phases to recommend suitable items to a given user in such situations:

1. Setting features/preferences: the user sets the features or preferences of items that she is looking for. Such features can be fixed explicitly or by means of examples [14]. From these features the system computes a user’s profile, $P_u$.

2. Filtering products/items: the system compares the user’s profile, $P_u$, with the items, $a_i \in A$. This filtering process is carried out by using similarity measures [39].

3. Recommendations: the most similar item or the top-N most similar items, $a_i \in A$, to the user profile, $P_u$, are recommended.

Similarly to the collaborative module, the knowledge based one must be modified to include the location-aware property. This modification is carried out in the filtering process as follows:

2. Location-aware knowledge based filtering: the comparison process between the user’s profile and the items considers just those items inside
the area defined by $R_{outer}$ and the user’s location, i.e., $a_i \in A'$. Figure 4 depicts this process.

3.2. Distance-based Re-ordering

Finally, we apply a contextual post-filtering module that re-orders the list of top-N recommended items provided by the previous phase using contextual information, in our case, the distance from the items to the user. For example, if two restaurants have similar predictions but one of them is much closer to the user than the other, it would be practical to recommend the closest on the farthest.

More formally, this module works as follows. Let $RL = \{ (a_i, v_i), \, i = 1 \ldots N \}$ be a recommendation list of $N$ items where $v_i$ is an estimation of the interest rate of the item $a_i$ for the target user. The distance-based re-ordering module receives this list as an input and modifies the interest rates $v_i$ according to the distance from each item $a_i$ to the user.

The new interest rate after the context-based re-ordering, $w_i$, is computed as:

$$w_i = v_i \cdot DD(u, a_i)$$ (1)
where $DD(u, a_i)$ is the distance decay for the item $a_i$ and the user $u$. The aim of this function is to reduce the interest rate of an item when the distance to the user increases.

Our proposal uses two parameters that define two concentric circular areas centered on the user’s current position, as depicted in Figure 5:

- **The outer radius** ($R_{outer}$) : This radius was introduced in section 3.1.1, and defines an interest area around the user. Outside this area, the distance to the item is considered excessive. Therefore, all items located out of this area are ignored and will not be recommended by the system.

- **The inner radius** ($R_{inner}$) : Within this circle, we consider that distances are low and thus, they have no influence in the interest rate of an item. The value of this parameter will be a fraction of the outer radius.

According to the proposal of Yang et al. [72], we use a negative exponen-
tial function to decrease the interest rating depending on the distance. The conditions for the proposed decay function are:

a) Negative exponential function: \( DD(u, a_i) = e^{-\lambda \text{dist}(u, a_i)} \), being \( \lambda \) a parameter that express the user’s distance sensitivity and \( \text{dist}(u, a_i) \) the Euclidean distance from the user \( u \) to the item \( a_i \).

b) For \( \text{dist}(u, a_i) < R_{\text{inner}} \) we have that \( DD(u, a_i) = 1 \)

c) For \( \text{dist}(u, a_i) \geq R_{\text{outer}} \) we have that \( DD(u, a_i) = \epsilon \), \( \epsilon \) being a sufficiently small positive number.

Therefore, the proposed distance decay function is defined as follows (see Figure 5):

\[
DD(u, a_i) = \begin{cases} 
1 & \text{if } \text{dist}(u, a_i) < R_{\text{inner}} \\
\frac{-\ln \epsilon}{R_{\text{outer}} - R_{\text{inner}}} e^{-\lambda \text{dist}(u, a) - R_{\text{inner}}} & \text{if } R_{\text{inner}} \leq \text{dist}(u, a) < R_{\text{outer}} \\
0 & \text{if } \text{dist}(u, a_i) \geq R_{\text{outer}} 
\end{cases}
\]

where the value of \( \lambda \) to satisfy the aforementioned conditions is given as:

\[
\lambda = \frac{-\ln \epsilon}{R_{\text{outer}} - R_{\text{inner}}}
\]

Finally, the result of this filtering will be a recommender list \( RL' = \{(a_i, w_i), i = 1 \ldots N\} \), sorted by \( w_i \) in descending order.

3.3. Enhancing Mobile Recommender Systems with 3D Maps

One important challenge of our RS is its user interface. We pursue the following goals:

1. The mobile application should be easy to use and driver-friendly.
2. It should be able to provide a spatial and intuitive representation of the user’s surroundings and the location of the recommended items.

3. It should run efficiently on thin mobile devices.

These goals were accomplished by integrating the RS engine described in the previous Section into a 3D GIS client-server technology, see Section 2.3. The 3D GIS allows tourists to visualize on their mobile devices a virtual 3D
representation of the world where they are *physically* located. Recommended items are portrayed on this 3D map according to the user location on the real world. See Figure 6 for some actual snapshots of our application running on an iPhone smartphone. As depicted, the map is rendered under a perspective projection, allowing users to visually search over entire cities or areas even on small mobile displays. Consequently, tourists can become familiar with the zone they are visiting and form an accurate cognitive map of the area around them. As well, 3D virtual representations are, in general, easier and faster to understand than 2D abstract maps [49].

The immersive user experience is further enhanced by matching the virtual view with the user’s current view in the physical world according to his/her location and view direction. These values are obtained, respectively, by the GPS receiver and by the electronic compass included on most current mobile devices. This automatic movement scheme resembles a car navigation system, and reduces the user interaction required to use the system. Alternatively, users can also interactively control the navigation according to a *walking metaphor* scheme [18]. This scheme allows tourists to move around their geographical space to locate items/areas of their interest using the device’s keyboard or touchscrreen.

The 3D representation provided by the mobile device is augmented by adding several landmarks. We identify two types of landmarks:

- **Recommended items.** The recommendations inferred by our hybrid recommendation engine described in Section 3.1 are used to populate the 3D environment. These landmarks are labeled with the name of the item and its recommendation value, see Figure 6. We use a 5-stars scale
to portray the recommendation values, as it is widely used by most recommender systems and known by most users. Venues, restaurants, cafés, pubs, bars, monuments, accommodation etc. are some examples of items that could be recommended using our proposed solution.

- **Points of interest.** POIs provide additional georeferenced information that can be of interest to users visiting the aforementioned recommended items, e.g., cash machines. Geographical features (e.g., mountains) are also depicted in order to provide an easier navigation and orientation.

However, a serious issue of mobile devices is their limited storage capacity. To overcome such a limitation, we employ an out-of-core rendering system which only stores a small subset of available 3D map into the device’s memory. The mobile device is in charge of downloading through a cellular network a simplified 3D representation of the terrain geometry from a remote GIS server. This representation is computed according to a custom multiresolution data structure described in [46, 47]. The map is progressively updated according to the customer movements across the environment, downloading new parts of the terrain from the server as they are needed.

4. **Architecture Design and Implementation**

We have developed a fully operational mobile recommender system that implements the proposed recommendation approach. This Section introduces our prototype whose goal is to provide recommendations to users about restaurants in the province of Jaén, Spain. This Section also describes the ar-
Figure 7: SR-REJA-3D

We propose a three-tier client-server architecture, as illustrated in Figure 7. It consists of three software components, namely, the Recommender Server, the GIS Server and the Mobile Client Application. Following subsections will describe each component and their interactions.

4.1. The Mobile Client Application

This application applies progressive downloading and rendering of 3D maps over cellular networks, such as GPRS and UMTS, as described in Section 3.3. It is also in charge of tracking the user’s location and speed based on GPS/compass and requesting recommendations from the server.

Cellular networks usually suffer from high latencies and low bandwidth. Therefore, in order to communicate efficiently the mobile clients with the GIS architecture and discusses issues related to the design and usage of our specific implementation.
Server under such limited conditions, we use a lightweight binary request-
response protocol built directly on top of TCP/IP [53].

Our client application has been developed as a plain C++ native program
using the industry-standard 3D graphics library OpenGL ES [35]. These
software tools guarantee a maximum efficiency on mobile hardware and an
optimum software portability [47]. Currently, it supports a broad range of
devices, including iOS (iPhone, iPad), Symbian OS, Windows Mobile, Win32
and GNU/Linux.

4.2. The Server

The server has been designed as two different applications: the GIS Server
and the Recommender Server, see Figure 7. These servers can be hosted on
the same machine or run as standalone applications on different computers.

The GIS Server manages all the communication with the mobile clients,
and provides them with data on demand. It is also in charge of storing
the complete terrain dataset. In our experiments, we have used a dataset
that covers the provinces of Jaén, Granada and Almería (Spain), approx.
41943 \( km^2 \), with a height spacing of 10m and a texture resolution of 5m per
pixel.

On the other hand, the Recommender Server implements the hybrid
model presented in Section 3.1 and applies it to restaurants. This scheme
makes our server capable of providing valuable recommendations according
to the user’s current location.

Contrarily to the client-server communication, the data interchange be-
tween the GIS and the Recommender Server is carried out by a fast wired
link. In this case, we employ a web service provided by the Recommender
Server, which allows XML messaging over the HTML protocol.

4.3. Recommendation Process

A typical scenario of user interaction might be as follows. First, a tourist logs in the system via his/her user name. Then, the mobile application establishes a network connection to the GIS Server, which will remain open until the application is closed. Next, the user’s current location is obtained via GPS and provided to the GIS Server. In response, an interactive 3D map is progressively streamed to the client based on this geographical position, see Figure 8a. The GIS Server also provides the client with different categories of POIs (cities, monuments, geographic features, etc.), which help users to match the physical and the virtual world while providing an easier navigation.

If the user requests a recommendation service, he/she may provide to the system some knowledge about his/her preferences, as described in Section 3.1.3. This step is mandatory if the system is under a cold start situation. In either case, the user can also explicitly provide the distance he is willing to travel (that is, the value of the parameter \( R_{outer} \) defined in Section 3.2). See Figure 8b. A request is then sent to the GIS Server with the user’s instantaneous location.

Once the request reaches the GIS Server, it is passed to the RS Server. Then, the recommendation engine generates a list of possible interesting restaurants based on the user’s profile and his/her preferences. These restaurants are later filtered according to the user’s context information, as described in Section 3.2. The resulting list of recommended items is returned to the GIS Server, and eventually sent to the client, which will use it to populate the 3D environment with landmarks, see Figure 8a. Apart from this
Figure 8: Actual screenshots of our application running on an iPhone. a) 3D map showing some recommended items. b) User preferences screen. c) A list with all the recommended items is also provided to the users. d) Restaurant details screen showed after selecting a restaurant.
3D interface, the application also provides users with a standard list of items arranged by recommendation value as depicted in Figure 8c. This list also includes some basic information about the restaurants, e.g., distance from the current user position to each restaurant.

The user can freely roam across the 3D environment, browsing the recommended items. Periodically, the mobile application calculates the Euclidean distance between the current location and the location where the last recommendation request was issued to the server. Whenever this distance exceeds the threshold value $R_{inner}$ defined in Section 3.2, a new recommendation request is issued to the server. That is, the set of recommendations showed to the user is constantly updated in real-time as the user location varies.

Once the user selects a restaurant (by touching on its landmark or by selecting it from the arranged list), he or she is able to review further descriptive information about such restaurant. Also, he/she is given the option to return to the 3D map, or to cast a rating, as showed in Figure 8d. User ratings are issued to the Recommender Server and saved in the user’s profile.

The recommendation process is illustrated in an accompanying video (see Figure 9) that shows a live recording demonstration of our prototype.

5. Evaluation

Evaluation of RS is usually performed using offline experiments and measured in terms of precision and accuracy in the recommendation [65]. However, this type of evaluation presents some limitations [8], for example, contextual conditions are difficult to emulate in an offline environment. Therefore, and following similar works on CARS and tourist guides for mobile
devices [8, 23, 58, 61], we have conducted a user study to evaluate our system.

We have built a fully operational prototype that implements the ideas described in previous sections, and it has been subjected to empirical experimentation. The goal of these experiments was to answer the two following questions, which reflect the main contributions of the paper:

1. Are 3D maps a good way to provide location-aware recommendations on mobile devices?
2. Does location-awareness increase the quality and usefulness of the system?

In order to assess the aforementioned questions, two additional variations of our mobile guide have been implemented:

- The first variant offers the same features, but the 3D map has been replaced by a standard 2D map, provided by Google Maps. See Figure 10a.
Figure 10: Screenshots of the 2D variants used in our experiments. The blue dot indicates the location of the user. Red pins are recommended restaurants. a) Location-aware version with \( R_{outer} = 2\text{km} \). b) Non location-aware version.

- The second variant is a non context-aware version. It behaves like a classic RS, i.e., user location is not taken into account in the recommendation process. See Figure 10b.

These two additional versions will be used as a baseline to compare and validate our proposal.

5.1. Usability Tests

We recruited 27 testers, 19 males and 8 females, with ages ranging from 24 to 48 years, being 30 the average age. Testers presented a good technological background. Most participants owned a wifi-enabled smartphone (87%). Also, they claimed to regularly access the Internet from their mobile device (72%). However, the amount of users that affirmed to be experienced in 3D
The experiments took place in the city of Jaén. All selected users were locals or have been living in the zone for some time. Therefore, they were familiar with the geography and restaurants of the test area. Prior to the experiment, evaluators were asked to create a personal profile using the web version of REJA, and to train the system by evaluating ten restaurants.

The user interface and the location-awareness have been studied independently. First, we conducted a user study to compare the proposed 3D user interface with a standard 2D solution, whereas the recommendation engine remained fixed. After that, we carried out a second experiment comparing the location-aware and the non location-aware versions, both using the 3D same interface.

In order to perform both evaluations, testers were asked to complete the following list of tasks:

1. Find a restaurant while walking in the campus of our university.
2. Find a restaurant while walking in the historic center of Jaén.
3. Find a restaurant while driving along a motorway.

Note that except from the first task, the contextual situations described in the previous list were simulated due to resource limitations. That is, users were requested to imagine they were in these situations. The system was adapted to provide recommendations according to these locations.

During our first experiment, evaluators could experiment with the 3D and the 2D variants of our system, in random order. After a brief explanation of the application, they proceeded to perform the aforementioned tasks. Upon completing them, evaluators filled out a usability questionnaire once per
tested variant. Following, we reproduce the questionnaire, which included some questions from [38].

Q1 It was easy to use this system.
Q2 The information provided by the system is clear and easy to understand.
Q3 I think that the user interface is appropriate for being used aboard a car.
Q4 I think that the user interface is appropriate for being used while walking.
Q5 I think that this system is of good practical use.

For the second experiment, the context-awareness of the system was turned off, and testers had to repeat the same tasks. After finishing, their evaluation and impressions were collected using the following questionnaire:

Q6 I’m satisfied with the restaurants suggested by the non location-aware version.
Q7 I’m satisfied with the restaurants suggested by the location-aware version.
Q8 Taking into account my location increases the usefulness of the recommendations.
Q9 I understood the benefit of using my location to suggest recommendations.
Q10 It was easy to set the maximum distance I’m willing to travel ($R_{outer}$).
Q11 I found it useful to adapt recommendations to the distance I’m willing to travel.
The questionnaires were answered using a seven-point [45] Likert scale where 1 means “strongly disagree” and 7 means “strongly agree”. Each question also had a free-text space for additional comments.

After the experiments, we proposed the testers to directly compare the evaluated versions with the following questions:

Q12 I prefer a 3D map over a 2D map.
Q13 I prefer the location-aware version over the non location-aware one.

Figures 11 and 12 collect the subjective ratings obtained from our usability tests. Also, 75% of the testers preferred the 3D map (Q12) and 100% of them preferred the location-aware version (Q13).
5.2. Discussion

Three dimensional maps provide users with more freedom of movement than 2D maps, which can reduce the efficiency of navigation [49]. Also, Google Maps is a very popular tool, and most evaluators were already trained in its use. Nevertheless our experiments proved that the proposed 3D interface was not found difficult to use. In fact, testers have expressed that both versions were equally easy to use (Q1), clear to understand (Q2) and have a good practical use (Q5). These results confirm our initial hypothesis that 3D maps are a valid tool for providing location-aware recommendations on mobile devices.

We observed, however, that the efficiency of the studied GUIs depends on the user’s mean of transportation (car or walking). The 3D GUI is clearly the preferred option to be used aboard a car (Q3). This stems from the fact that it provides a more natural and practical view of the road. On the contrary, the 2D variant received a slightly better consideration for the walking scenario (Q4). This result attends to the following reasons: our virtual representation did not include 3D buildings; cities do not usually present significant geographic features; and the testers found 2D maps more akin to classic street maps.

Therefore, 2D and 3D maps are complementary GUIs with different fields of application. Nonetheless, 3D maps achieved good qualifications in both scenarios, proving to be more polyvalent that their 2D counterparts. Note that a 3D map can perfectly emulate a 2D one by just pointing down the virtual camera to the ground and adjusting the altitude (zoom) adequately. This is the default view used by Google Earth.
Concerning the second questionnaire, see Figure 12, evaluators showed a clear preference for the proposed location-aware system over the standard variant. Comparing the user satisfaction expressed in Q6 and Q7, we observe that the evaluation were very positive for both versions, although the location-aware system was substantially more satisfying. Questions Q8 and Q9 answer the second question formulated at the beginning of this section: our experimentation proved that location-aware recommendations were perceived more useful than non location-aware ones by most users.

Still, we should point out that several evaluators claimed that under certain circumstances, they would prefer travelling longer in order to go to the best possible restaurant. These evaluators appreciated the option to configure $R_{outer}$, as it gives them the opportunity to increase or reduce the importance of the context-aware part of the system at their will. In consequence, this option was uniformly considered both an easy (Q10) and efficient (Q11) way to achieve a good trade-off between distance and quality of the restaurants.

Finally, the two ending questions (Q12 and Q13) served as a summary to our user study. When the evaluators were asked to compare the proposed 3D system with the baseline 2D variant, most of them preferred the 3D version (Q12: 75%). This result seems to indicate that, in general, users felt very attracted to this new GUI, although there is still room for improvements. On the other hand, the evaluation demonstrated that the location-aware variant was strictly preferred regarding the non-location aware one (Q13: 100%).

In addition, evaluators also requested various additional features and provided some critical comments, which will be considered for future versions of
the application. Following, we summarize the evaluators’ feedback:

- **Routing functionalities.** Users claimed that they wanted the application to guide them to the selected restaurant.

- **Recommendation of POIs along the route and the destination point,** not only around the current location.

- **Inclusion of additional layers of vector information on the 3D map** (roads, streets, etc.).

- **Possibility to switch from the 3D to the 2D interface and vice versa.**

- **Capability to add comments about restaurants and review comments provided by other users.** Integration with social networks.

The first two points, coupled with the good performance obtained by the 3D map version in the car scenario, indicate that context-aware recommendation technologies are an interesting solution for car navigation systems, and that drivers are demanding smart navigation systems to assist them in their travels.

6. **Concluding Remarks**

The new generation of smartphones and tablets include features which were unconceivable some years ago: surprising graphics capabilities, built-in sensors and ubiquitous connection to the Internet. This paper presents a novel recommender system that takes full advantage of these features. Our
Our solution adapts the recommendations provided to users according to their current physical location. Moreover, it also employs an intuitive 3D map-based interface that overcomes the limited display size of most mobile devices, as well as presents a rich and detailed virtual representation of the world where the tourists is currently located. Our system fulfills the basic necessities required for on-the-move tourists: where I am, what interesting items can be found nearby, how far I am from them and how do I reach them. The details related to the design and implementation of a client-server application that implements these ideas were also provided.

Finally, the proposed solution has been implemented and an evaluation with testers has been carried out in order to validate our ideas. Testers have provided a very positive review of the application, pointing out the simplicity of its use and the usefulness of the real-time location-based recommendations.

Acknowledgments

This work has been partially supported by the Ministerio de Ciencia e Innovación (Spain) and the European Union (via ERDF funds) through the research projects P07-TIC-02773 and TIN2009-08286.

References

421–433.


39


[45] G. Miller, The magical number seven, plus or minus two: Some limits on our capacity for processing information, 1956.


[59] R. Rodríguez, M. Espinilla, P. Sánchez, L. Martínez, Using linguistic
incomplete preference relations to cold start recommendations, Internet

[60] A. Said, Identifying and utilizing contextual data in hybrid recommender
systems, in: Proceedings of the fourth ACM conference on Recom-
365–368.

[61] N. Saiph Savage, M. Baranski, N. Elva Chavez, T. Hllerer, I’m feeling loco: A location based context aware recommendation system, in:
G. Gartner, F. Ortag (Eds.), Advances in Location-Based Services, Lecture Notes in Geoinformation and Cartography, Springer Berlin Heidel-

[62] J. Schafer, J. Konstan, J. Riedl, E-commerce recommendation applica-

[63] W. Schwinger, G. Ch, B. Pröll, W. Retschitzegger, A. Schauerhuber,
Context-awareness in Mobile Tourism Guides, Technical Report, Institu-
tute of Bioinformatics JKU Linz, 2005.

[64] M. van Setten, S. Pokraev, J. Koolwaaij, Context-aware recommenda-
tions in the mobile tourist application compass, in: P. De Bra, W. Nejdl
(Eds.), Adaptive Hypermedia and Adaptive Web-Based Systems, vol-
ume 3137 of Lecture Notes in Computer Science, Springer Berlin / Hei-


