

TourismShare

Nuno Areias¹ and Benedita Malheiro^{1,2}

¹ ISEP/IPP – School of Engineering, Polytechnic Institute of Porto, Portugal

² INESC TEC, Porto, Portugal
{1090430,mbm}@isep.ipp.pt

Abstract. TourismShare is a context-aware recommendation platform that allows tourists to share private locations and videos and obtain recommendations regarding potential Points of Interest (POI), including complementary articles and videos. The user experience is enhanced with the addition audio immersion during video playback and automatic recommendation features. The developed system consists of a distributed application comprising a front-end client module (Android application), which provides the user interface and consumes directly external support services, and the back-end server module, which includes the central database and recommendation service. The communication between the client and server modules is implemented by a dedicated application level protocol. The recommendations, which are based on the user context (user position, date and current time, past ratings and user activity level), are provide on request or automatically, whenever POI of great relevance to the user are found. The recommended POI are presented on a map, showing the timetable together with complementary articles and videos. The audio immersion at video playback time takes into account the weather conditions of the video recording and the user activity level.

Keywords: Context-aware, Recommendation, Immersion, Tourism.

1 Introduction

Information and Communication Technologies (ICT) have greatly contributed to the proliferation of data sources and resource sharing systems. This abundance of information increases the importance of filtering tools such as recommendation systems, which are able to make timely customized suggestions based on the user profile and current context.

The main objectives of this project are: (i) to allow the user to share POI and videos; (ii) to recommend POI, articles and videos based on the user's context; and (iii) to provide the user with a richer video playback experience, simulating the recording conditions. This system was inspired by two previous location-based sharing and recommendation projects: (i) the POI sharing and recommendation of Bruyneel *et al.* (2014) [3]; and (ii) the video sharing and article recommendation of Baecke *et al.* [1]. The developed system provides POI recommendations together with complementary articles, videos and timetable, based on the user context information (user position, date and time, past ratings

and activity level) on request or automatically. The user immersion at video playback time takes into account the time and weather conditions of the video recording. Specifically, TourismShare improves the tourist experience not only by recommending context-aware POI, articles and videos, but also by allowing the user to decide when and which POI and videos to share and by enriching video playback with audio immersion.

This paper is organised in four sections. This initial section introduces the problem, the goal and the structure of the paper. The second section provides the background on context-awareness and immersiveness. The third section describes the TourismShare system. Finally, the fourth section draws the conclusions and suggests future developments.

2 Background

Recommendation systems are information filtering systems which suggest items or predict the ratings a user would attribute to items based on his/her profile and context. Recommendation systems are becoming increasingly popular since they filter large quantities of data to find and suggest items of interest to users. Nowadays, it is possible to find recommendation systems in e-commerce, e-health, tourism or technology enhanced learning application domains [6,3,12].

2.1 Context-awareness

Context includes any information relating to the user which is relevant to the purpose of the application, including demographic, location, time, preferences or social data. A context-aware recommender system predicts user preferences by incorporating contextual information into the recommendation process [4] and allowing it to generate more relevant recommendations [5]. The rating depends on the user, item and context, where the context can be the combination of different types of contextual information like time, location or demographics.

In terms of contextual information, TourismShare uses the current user position, activity level and time. The position is used to filter the POI based on the distance between the user and POI, excluding all POI outside the maximum distance radius circle. The maximum distance depends on the current user activity level. The activity level, which is derived from the average user velocity during the last 5 min, is expressed in a 0 to 5 scale corresponding to unknown, inactive, low activity level, medium activity level and high activity level. The time is used to recommend, based on the opening hours, only open POI.

2.2 Immersion

Immersion is a subjective experience of total involvement in a given environment and that can be suggested by using surround sound sensory experiences, high resolution and augmented reality [10]. The usage of immersion features such as different media elements highly enriches user experience by creating emotional

reactions. An example of the usage of such features is the video immersion system proposed by Ramalho *et al.* (2013) [9], which allows users to capture, search, publish, view and synchronize interactive TV items. During the video capturing process the system collects information in the form of metadata of the user position, velocity, orientation and current weather information. During video playback, this information is used to implement different perception visual, auditory and tactile features. We adopt a similar approach to implement audio immersion during video playback based on the collection of weather information – used to reproduce weather sounds – and smartphone sensory data – used to reproduce a soundtrack matching the user activity level. The application selects the soundtrack from a play list with soundtracks matching the pre-defined levels of user activity, *i.e.*, from classical music (when the user is inactive) to rock/metal (when the user is highly active).

2.3 Related Work

Table 1 shows the comparison of the six tourism location-based context aware recommender systems which were studied: ErasmusApp [3,1], Cinemappy [7], MoreTourism [11], Rerex [2], SPLIS [13] and CAMR [8].

Table 1. Comparison of related systems

Features	ErasmusApp	Cinemappy	MoreTourism	Rerex	SPLIS	CAMR
Position	✓	✓	✓	✓	✓	✓
Time	✗	✓	✓	✓	✓	✓
Company	✗	✓	✗	✓	✗	✗
Weather	✗	✗	✗	✓	✓	✓
Crowdedness	✗	✗	✗	✓	✗	✗
User Mood	✗	✗	✗	✓	✗	✗
Temperature	✗	✗	✗	✓	✗	✓
Activity	✗	✗	✗	✗	✗	✓
Lighting	✗	✗	✗	✗	✗	✓
Noise	✗	✗	✗	✗	✗	✓
Immersion	✗	✗	✗	✗	✗	✗

This analysis shows the relevance of this project. Although some of these systems implement complex context-aware recommendation algorithms and rely on high volumes of contextual data, none enriches the user experience with immersive features.

3 TourismShare

TourismShare is a distributed application composed of three main modules: front-end, back-end and external services (Figure 1). The front-end module provides the user interface and includes a local (SQLite) database to store private locations and messages. The back-end module manages the user access to the central database and generates recommendations when requested. The communication between the client and server modules is achieved through a dedicated

application level protocol. The external services provide map rendering, video upload and playback, weather data and complementary Wikipedia articles.



Fig. 1. TourismShare architecture

3.1 Back-end

The back-end provides two services: persistent data storage and personalised recommendations. The central database is a MySQL database composed of multiple tables, including User, Publisher, Rating, Location, Subcategory, Category, Schedule, LocationSchedule, Report, ReportCat, Device, UserDevice, Videos, WeatherData, Music and several tables for the smartphone sensor data. The recommendation service is based on the available data: (i) spatial information – geodetic coordinates of the user; (ii) temporal information – date and time of the search; (iii) user information – level of activity; and (iv) device information – brand, model, size, *etc.* The recommendation algorithm is a cascade of four filters governed by the following rules: (i) a filter is only applied if the number of input items is above five; and (ii) the outcome of a filter is only used if the the number of output items is above three. In the latter case, the recommendation service uses the outcome of the previous filter.

The first filter – the pre-filter – is used for the recommendation of POI and videos. It collects all items which satisfy the following criteria: (i) were published by other users (mandatory); (ii) are located within a radius of 5 km from the search site (mandatory); (iii) match the category or subcategory (optional); and (iv) are new sites (optional). The second filter – the spatial filter – narrows the radius of search based on the user activity level: unknown or inactive – 0.75 km; low activity level – 1.50 km; medium activity level – 2.00 km; and high activity level – 3.00 km. The third filter – the temporal filter – extracts all sites which are open at the moment of the search. The last filter – the prediction filter – is

a rating prediction algorithm which returns the top five rated locations for the user. The filter considers three cases: (i) the user has previously rated the site; (ii) the user has never rated the site, but has rated sites of the same category; and (iii) the user has never rated the site nor sites of the same category.

If the user has previously rated the site, the algorithm, first, applies Equation 1 to determine the reputation of the site's publisher $Rep(p, l)$, where p is the publisher, l is the site, l_{pub} is the number of sites p published and l_{rep} the number of active reports associated with these sites. Users, while site publishers, have a dynamic reputation, ranging from 0 % and 100 %. Secondly, uses Equation 2 to calculate the average rating of the site l of the category c , where n is the number of users who rated the site and $Rating(u, l, c)$ is the rating user u assigned the site. Finally, applies Equation 3 to predict the user rating for the site $\widehat{Rating}(u, l, c)$, where u represents the user, $Rating(u, l, c)$ is the previous rating the user assigned to the site, $Rep(p)$ is the reputation of the site publisher, $\overline{Rating}(l, c)$ is the average rating of the site and $\beta = 80$ %.

$$Rep(p, l) = \frac{l_{pub} - l_{rep}}{l_{pub}} \quad (1)$$

$$\overline{Rating}(l, c) = \frac{1}{n} \sum_{u=1}^n Rating(u, l, c) \quad (2)$$

$$\widehat{Rating}(u, l, c) = \beta Rating(u, l, c) + (1 - \beta) Rep(p) \times \overline{Rating}(l, c) \quad (3)$$

If the user has never rated the site, but has rated sites of the same category, the algorithm first applies Equation 4 to determine the average rating user u has given to sites of category c , where n is the number of sites of category c the user has rated and $Rating(u, l, c)$ is the rating user u assigned the site l of category c . Then, uses Equation 5 to predict the site rating for the user $\widehat{Rating}(u, l, c)$, where $\overline{Rating}(u, c)$ is the average rating user u has given to sites of category c , $Rep(p, l)$ is the reputation of the site publisher given by Equation 1, $\overline{Rating}(l, c)$ is the average rating of the site given by Equation 2 and $\beta = 60$ %.

$$\overline{Rating}(u, c) = \frac{1}{n} \sum_{l=1}^n Rating(u, l, c) \quad (4)$$

$$\widehat{Rating}(u, l, c) = \beta \overline{Rating}(u, c) + (1 - \beta) Rep(p, l) \times \overline{Rating}(l, c) \quad (5)$$

If the user has never rated the site nor sites of the same category, the algorithm applies Equation 6 to predict the site rating for the user $\widehat{Rating}(u, l, c)$, where $Rep(p, l)$ is the reputation of the site publisher given by Equation 1, $\overline{Rating}(l, c)$ is the average rating of the site l given by Equation 2 and $\beta = 40$ %.

$$\widehat{Rating}(u, l, c) = \beta \overline{Rating}(l, c) + (1 - \beta) Rep(p) \times \overline{Rating}(l, c) \quad (6)$$

In this case, the weight of the publisher is higher since the active user does contribute with any input to the rating prediction.

In the case of video recommendation, the algorithm applies an additional filter to select popular videos based on the ratio between the number of views

and the time elapsed since the date of the video publication. New videos start with a default ratio of 0.3. Finally, the video soundtracks are selected according to the user activity level.

3.2 External Services

The Android mobile application interacts directly with Google Maps, GeoNames, OpenWeatherMap and the YouTube services. The Google Maps service provides a map-based graphical user interface showing the user position and the various private or recommended POI. The interaction with the Google Maps service, including connecting, loading, map displaying and interacting, is supported by the Google Maps Android API and requires authentication. The GeoNames service provides Wikipedia articles related to the area surrounding the user location. The interaction is supported by the GeoNames API. The service requires authentication and provides a maximum of 40 English articles related to POI located in the region centred on the user location and with the specified radius. The OpenWeatherMap service returns the weather of a location. The application accesses the service during video recordings to obtain the meteorological information, *e.g.*, humidity, pressure, wind or temperature. The YouTube service requires authentication and allows video storage and sharing. The application uses the YouTube Direct Lite API to interact with the YouTube service [1].

3.3 Front-End

The front-end comprises the client Android application and the local database and interacts with external services and the back-end. The local SQLite database is composed of the Location, Subcategory, Category, Message and MessageCat tables. The client application includes three independent Android services: (i) the MyPVTService, which is responsible for obtaining the position, velocity and time (PVT) of the user; (ii) the RecommendService, which is responsible for the automatic recommendation sites, videos and articles; and (iii) the MusicService, which is responsible for the audio immersion. These services are launched at start-up and remain active until they are disabled by the user in the application settings (RecommendService and MusicService).

MyPVTService provides the user's PVT. This event driven service, whenever there is a location-related event, verifies the source and the specified data refresh period. While the network provider has a refresh period of 90 s, the GPS sensor updates every 30 s. The GPS sensor events are granted higher priority because they have greater accuracy. Regardless of the source (GPS sensor or network). If the refresh period had been exceeded (30 s or 90 s), the service collects new information. The velocity is stored in a First In, First Out (FIFO) buffer queue of size 10 used to establish the user's activity level. If the update period is higher than 20 min, the service classifies the activity level as unknown.

RecommendService supplies POI recommendations. This service, every 5 min, calculates the distance between the current and the previously recorded user location, assuming the user is authenticated, has the automatic recommendation mode on and is connected to the Internet, If this distance is greater than 300 m, it invokes the back-end recommendation service, avoiding generating redundant recommendations. Whenever new recommendations are generated, the system checks and only notifies the user of the new items.

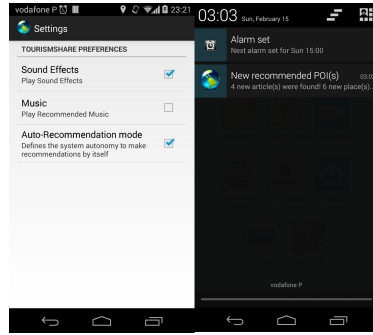
MusicService is responsible for the audio immersion. Assuming the user is authenticated and has access to the Internet, the application requests the playlists to the back-end server. If the user's activity level is unknown, the service checks periodically (every 3 min) for changes. Once the level of activity is known, the service starts playing the corresponding playlist. There are four playlists associated with the four pre-defines levels of activity (unknown, low, medium, high). The audio immersion, once started, rechecks at the end of each song the user level of activity and, depending on the outcome, may start playing another list.

3.4 Features

TourismShare has two operation modes: stand-alone and connected. In stand-alone or private mode, the user can only use a subset of the system features since the front-end application is disconnected from the back-end. They include user authentication, user registration, user data update, private POI edition and private POI rating. In connected or shared mode, the front-end interacts with the back-end, requiring successful user authentication. It provides access to the full set of implemented functionalities, including user authentication, user registration, user data update, POI edition, POI rating, POI sharing, POI reporting, video playing, video recording & sharing, on-request POI recommendation and automatic POI recommendation.

Automatic Recommendation The user is notified every time a new POI is recommended (Figure 2a). This feature can be enabled/disabled in the application settings.

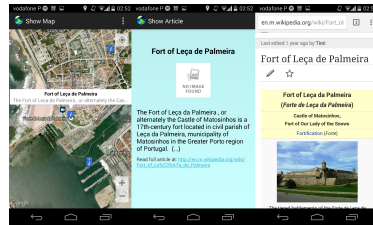
On Request Recommendation The user can request POI recommendations at any time and specify: (i) the type(s) of POI (places, articles, videos); (ii) the category and subcategory of the places; (iii) the time (current or another); and (iv) the location (current or another). After submitting the request, the user is redirected to a map of his surroundings showing the recommended POI with different markers depending on the type of POI and, in case of a place, the category and subcategory (Figure 2b). When the user clicks on an article marker, a new screen displays a brief of the Wikipedia article together with a hyperlink to the full article (Figure 2c). If the user clicks on a place marker, a menu appears for the user to: (i) report a location; (ii) rate a location; (iii) add a location to the list of favourites; and (iv) show a location schedule (week/weekend) (Figure 2d).



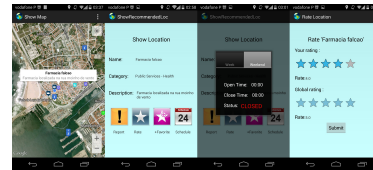
(a) Automatic recommendation



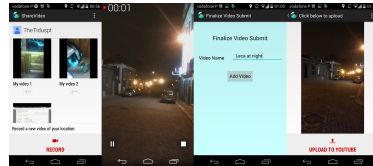
(b) On request recommendation



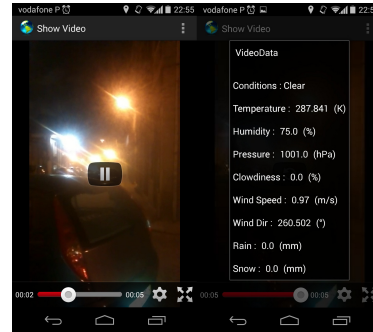
(c) Show recommended article



(d) Show recommended location



(e) Video sharing



(f) Audio immersion

Fig. 2. TourismShare Features

Video Sharing This feature uses YouTube and, thus, requires the user to be logged into a Google account. During the recording process, the application collects the weather and sensor data which will be used for the audio immersion during the video playback. After recording, naming and reviewing a video, the user can upload it to Youtube (Figure 2e).

Audio Immersion During video playback, the application accesses the weather information gathered at the time of the video recording and the current activity level of the user. It plays a background wind sound and a soundtrack according to the current user activity level and the wind velocity at the

time of recording. There are four playlists, one for each user activity level (stopped, low, medium, high). Figure 2f illustrates the video immersion. The user can enable/disable this feature in the application settings.

3.5 Tests and Results

In order to characterize the execution of the different features, a performance test was made under the following conditions: *(i)* the front-end device was a Motorola Moto G with a Central Processing Unit (CPU) Quad-core 1.2 GHz Cortex-A7, 1 GB random-access memory (RAM) and accelerometer and proximity embedded sensors; *(ii)* the back-end platform was Compaq Pressario laptop with a AMD Sempron M120 / 2.1 GHz CPU and 4 GB RAM; and *(iii)* an optical fibre Internet connection which, during the data transmission test, presented an average download rate of 19.08 Mb/s, an average upload rate of 1.67 Mb/s and an average latency of 81 ms. Table 2 displays the obtained results.

Table 2. Average elapsed time, upload and download data per feature

Feature	Time (s)	Upload (B)	Download (B)
User authentication	1.33	856	658
User registration	1.77	1144	660
User data update	1.27	1053	663
Rate place	1.71	774	653
Report place	1.65	783	578
Share place	1.91	1274	664
Edit place	1.88	1235	658
Play video	1.21	795	576
Share video	2.55	3764	576
On-req. POI recomm.	2.01	1124	4367
Autom. POI recomm.	2.32	1174	4567

During the test, the average elapsed time, the data exchanged between the client device and the server (upload) and between the server and the client device (download) for each feature was determined based on ten measurements.

4 Conclusions and Future Work

TourismShare is a tourism context-aware platform for Android smartphone users. In terms of contextual data, it uses: *(i)* the current location, level of activity, device and time of the user to make recommendations; and *(ii)* weather data and the current activity level of the user for audio immersion. TourismShare improves the tourist experience by recommending context-aware POI, articles and videos, by allowing the tourist to decide when and which personal data (POI and videos) to share and by enriching video playback with audio immersion.

In the future, the recommendation algorithm can be refined by using information it currently collects, *e.g.*, the device characteristics, and by integrating social network data. The video immersion system, which is currently based on

weather information, can be extended by using local sensor information already collected and stored. Finally, the current audio immersion, which is based on the user activity level, may evolve into a music recommendation system based also on the user emotional state and profile.

Acknowledgements

This work was partially financed by the European Regional Development Fund (ERDF) through the Operational Programme for Competitiveness and Internationalisation (COMPETE Programme), within project «FCOMP-01-0202-FEDER-023151» and project «POCI-01-0145-FEDER-006961», and by national funds through the Fundação para a Ciência e Tecnologia (FCT) - Portuguese Foundation for Science and Technology - as part of project UID/EEA/50014/2013.

References

1. Baecke, B.: Context-Aware Video-Sharing Android Application. Master's thesis, Instituto Superior de Engenharia do Porto, Portugal (2014)
2. Baltrunas, L., Ludwig, B., Peer, S., Ricci, F.: Context-aware places of interest recommendations for mobile users. In: CEUR Workshop Proceedings. vol. 740 (2011)
3. Bruyneel, K., Malheiro, B.: Erasmusapp: A location-based collaborative system for erasmus students. In: De Strycker, L. (ed.) ECUMICT 2014, Lecture Notes in Electrical Engineering, vol. 302. Springer International Publishing (2014)
4. Gediminas Adomavicius, Ramesh Sankaranarayanan, S.S., Tuzhilin, A.: Incorporating contextual information in recommender systems using a multidimensional approach. *ACM Transactions on Information Systems (TOIS)* 23(1) (2005)
5. Jong-yi Hong, Eui-ho Suh, S.J.K.: Context-aware systems: A literature review and classification. *Expert Systems with Applications* 36(1) (2009)
6. Melville, P., Sindhvani, V.: Recommender systems. In: Sammut, C., Webb, G.I. (eds.) *Encyclopedia of Machine Learning*, pp. 829–838. Springer US, Boston, MA (2010)
7. Ostuni, V.C., Noia, T.D., Mirizzi, R., Romito, D., Sciascio, E.D.: Cinemappy: a context-aware mobile app for movie recommendations boosted by dbpedia. In: CEUR Workshop Proceedings. vol. 919 (2007)
8. Otebolaku, A.M., Andrade, M.T.: Context-aware media recommendations for smart devices. *Journal of Ambient Intelligence and Humanized Computing* 6(1) (2015)
9. Ramalho, J., Chambel, T.: Immersive 360° mobile video with an emotional perspective. In: *Proceedings of ImmersiveMe 2013*. ACM (2013)
10. Ramalho, J., Chambel, T.: Windy sight surfers: Sensing and awareness of 360° immersive videos on the move. In: *Proceedings of ImmersiveMe 2013*. ACM (2013)
11. Rey-López, M., Barragáns-Martínez, A.B., Peleteiro, A., Mikic-Fonte, F.A., Burguillo, J.C.: moretourism: Mobile recommendations for tourism. In: *Proceedings of ICCE'2011*. IEEE (2011)
12. Verbert, K., N., M., Ochoa, X.W., M. Drachsler, H., Bosnic, I., Duval, E.: Context-aware recommender systems for learning: A survey and future challenges. *IEEE Transactions on Learning Technologies* 5(4) (2012)
13. Viktoratos, I., Tsadiras, A., Bassiliades, N.: Geosocial splis: A rule-based service for context-aware point of interest exploration. In: *Challenge+ DC@ RuleML* (2014)