



Cultural heritage visits supported on visitors' preferences and mobile devices

Pedro J. S. Cardoso¹ · João M. F. Rodrigues¹ · João Pereira¹ · Sergey Nogin² · Joana Lessa³ · Célia M. Q. Ramos⁴ · Roman Bajireanu¹ · Miguel Gomes⁵ · Paulo Bica⁵

Published online: 18 June 2019
© Springer-Verlag GmbH Germany, part of Springer Nature 2019

Abstract

Monuments, museums and cities are great places to feel and experience neat and interesting things. But cultural heritage is experienced differently by different visitors. The more erudite may know beforehand what they intend to explore, while the least literate usually know and are capable of expressing some of their preferences but do not exactly realize what to see and explore. This paper proposes the use of a mobile application to set an itinerary where you can move at your own pace and, at the same time, have all the complementary information you need about each of the points of interest. The application is designed in face of an adaptive user interface where the routing and augmented reality are connected to acknowledge the needs of different user categories, such as elders, kids, experts or general users

Keywords Adaptive navigation · Mobile application · Adaptive user interface · Cultural heritage router planner · Ant colony optimization · Multi-criteria optimization · Enhancing accessibility · Fighting info-exclusion

1 Introduction

Technology is changing the way cultural heritage is experienced. Traditional visits to museums, cities and other spaces include a predefined walk, or set of walks, which do not necessarily satisfy the majority of the users' real preferences and needs. Many times the number of points of interest (POIs) is

also large, making impossible to experience all of them in a limited time window, and therefore necessary to proceed to a careful selection of what is going to be explored. Furthermore, it is possible to imagine a course in a museum space where visitors can see, hear, feel, smell and maybe even taste what existed at the time when the musicological piece was developed, or even the pieces' contents [41, 46, 48]. Enhancing accessibility and fight info-exclusion is another vector which should be explored, using, for instance, some classification systems to include features which reflect the degree of impairment of the visitor along with its preferences. For example, painting POIs probably are not adequate for a blind

This work was supported by the Portuguese Foundation for Science and Technology (FCT) projects LARSyS (UID/EEA/50009/2013) and CIEO (UID/SOC/04020/2013), and project M5SAR I&DT nr. 3322 financed by CRESC ALGARVE 2020, PORTUGAL 2020 and FEDER.

✉ João M. F. Rodrigues
jrodrig@ualg.pt

Pedro J. S. Cardoso
pcardoso@ualg.pt

João Pereira
japereira@ualg.pt

Sergey Nogin
sergeyn91@gmail.com

Joana Lessa
jlessa@ualg.pt

Célia M. Q. Ramos
cmramos@ualg.pt

Roman Bajireanu
romanbajireanu@gmail.com

Paulo Bica
pbica@spic.pt

¹ LARSyS and ISE, University of the Algarve, Faro, Portugal

² ISE, University of the Algarve, Faro, Portugal

³ CIAUD and ESEC, University of the Algarve, Faro, Portugal

⁴ CIEO and ESGHT, University of the Algarve, Faro, Portugal

⁵ SPIC - Creative Solutions, Loulé, Portugal

person (unless some arrangements are implemented, e.g., see, [5]) or some predefined walks include stairs which are not transposable by people in wheel chairs or other mobility disability.

In parallel, traditional user interfaces (UI) follow a *one-size-fits-all policy*, ignoring the needs, abilities and preferences of individual users. Moreover, research indicated that visualization performance could be improved by adapting aspects of the visualization to the individual user [52]. Intelligent user-adaptive interface, adapting on the fly to the specific needs and abilities of each individual, is a long-term research goal [11]. The referred complexity rises from two main reasons: the difficulty of extracting information about the users needs and abilities, and the implementation of the UI that can adapt/change “itself” on the fly. However, the modular and/or adaptive generation of UI offers the promise of providing personalized interfaces on-the-fly, though this does not mean that the user will be satisfied with his/her personalized application. According to Zhao et al. [62], the psychological process behind satisfaction is highly complex and requires a differentiation between transaction-specific satisfaction and cumulative satisfaction. Nevertheless, mobile applications (Apps) should move toward completely personalized experiences. These experiences usually are built from the aggregation of many individual pieces of content, such as augmented reality (AR).

This paper focuses on the implementation of a router planner (and AR) App that uses collected information to build AUI, as a part of the work being developed in the Mobile Five Senses Augmented Reality System for Museums (M5SAR) project [41, 46, 48]. The proposed solution uses a multi-criteria formulation for the route recommender system [16]. Besides restrictions, as the maximum allowed time for the visit, three goals were devised with the objective of optimizing: (1) the user’s preferences, (2) the number of visited POIs and (3) the time spent exploring POIs. To solve the optimization problem, a set of methods to design the visit of the users through some space were designed. The methods are supported on the ant colony optimization [17] algorithms and a weighted function strategy. As input, the methods require a network with a set of POIs categorized according to some classification system and information from the user’s preferences over the same classification system. Results are presented for a network which represent a museum and for the network of Faro city (Portugal). In addition, the present solution, by using AR, allows adding preset signals (e.g., paintings, statues) easily detectable in the environment to improve the App’s performance and enhance users experience, but also helps in the user localization, that it is complemented by using a beacon system [18]. The main paper contribution is the combination between an AUI and the intelligent adaptive user routing for museums or other cultural places.

The remaining document is structured as follows. The next section further extends the AUI and the routing state of the art. Section 3 presents our routing formulation of the problem and describes the proposed methods, while some results are explored. In the fourth section, the integration of the AUI is presented, along with the routing results. The last section presents some conclusions and future works.

2 Contextualization and state of the art

The number of mobile Apps, including the ones that use AR, is increasing due to the democratization of mobile devices and their enlarging number of functions, such as built-in cameras and global positioning systems. The massive availability of Internet connections on mobile devices also enables the construction of personal context-aware cultural experiences [31]. In this sense, UI is a fundamental research area, where the core of the investigation in the near future should fall, most probably, in the usually called intelligent user interfaces (IUI) or adaptive user interfaces (AUI) and on the automatic generation of interfaces (AGI), connected with the best practices of interaction of design (IxD), user experience (UX) and emotional UI (EUI) [46].

Schuller [49] argues that what will differentiate future IUI is the commitment to lend them “emotional intelligence”: interfaces that realize and can react properly to the users’ pleasure or displeasure. The information is thereby increasingly accessed from multiple modalities, in affect recognition and sentiment analysis, thanks to the availability of increasingly large and realistic resources, including deep learning and long short-term memory architectures, and weakly supervised learning methods [36, 39, 49]. In fact, in the best of all worlds, the system would have sufficient knowledge about a users’ culture before their first accesses to the interface, because the first impression counts [34]. Reinecke and Bernstein [43] argue that, to appeal to users in expanding markets, a more comprehensive personalization of interfaces to the cultural background is needed. The authors identify ideas on how to obtain user information in order to subsequently adapt the UI to certain aspects, e.g., see [20, 33]. In this context, also very important is to adapt the UI to users with different visual, auditory or motor impairments. Unfortunately, because of the great variety of individual incapacities among such users, manual modular designing interfaces for each one of them are impractical and not scalable [20, 45], at least until the design is (semi-) automatically generated by AGI.

In terms of applications, a study about adaptive model-driven UI development systems, where the focus was AUI for mobile application applied to museums, was presented in [1]. In this work, an overview of adaptive model-driven UI development systems, including a set of criteria to evaluate

the strengths and shortcomings of these systems, is outlined. Karaman et al. [32] described a system called MNEMOSYNE that, supported on passive observation, builds a profile of the artworks of interest for each visitor. Gajos and Weld [19] proposed an automatic system for generating UI, a solution based on treating interface adaptation as an optimization problem. In order to approach accessible designs for mobile devices, the principles of universal design are important to be followed [8], as well as the guidelines for supporting accessibility [26]. Cardoso et al. [7] proposed a system that works with the users' actions to feed an a priori (machine learning) algorithm in order to suggest which objects to visit in a museum. In summary, mobile applications should move toward completely personalized experiences.

In terms of museum Apps, almost every great museum has its own, corresponding to a huge amount and variety of applications. For instance, The Wall Street Journal, the Information Week and The Balance present articles with their reviews of the best Apps for the visit of museums [3, 30, 61]. The use of AR in museums was investigated, e.g., by Vainstein et al. [55], including the implementation of head-worn displays, and requirements and additional features for usable AR systems in museums. Other AR solutions, nonexclusive for museums, are the Srbija 1914/Augmented Reality Exhibition at the Historical Museum of Serbia [51], the Invisible Museum from Qualcomm [42], the interactive IPAD Museum catalog from the University of Virginia Art Museum [54], the interactive devices at the Cleveland Museum which includes playing games and social interaction, with face and posture recognition [10], or the Science Museum - Atmosphere Gallery [50].

In parallel to the AUI subject, the routing based on users' preferences is also being studied for some time. The Rijksmuseum Amsterdam offers a real-time routing system that implements a mobile museum tour guide for providing personalized tours tailored to the user's position inside the museum and interests [56]. The system includes tools for the interactive discovery of user's interests, semantic recommendations of artworks and art-related topics, and the (semi-)automatic generation of personalized museum tours. Benouaret and Lenne [4] proposed a recommender system for mobile devices. The system adapts to the users' preferences and is sensitive to their contexts, building tours on-site according to their preferences and constraints. A state of the art in the field, proposing a classification of mobile tourism recommender systems and providing insights on their offered services, can be found in the work of Gavalas et al. [23]. The CHESS [9] project researches, implements and evaluates both the experiencing of personalized interactive stories for visitors of cultural sites and their authoring by the cultural content experts. Spatially broader, the Route Perfect [47] platform allows to easily plan a trip in Europe based on the traveler preferences, budget and personal style. Several

other works can be found in the literature such as [7, 21, 59, 60].

Following the authors' previous works in the pursuit of a full AUI [45, 46], a complete adaptive navigation module is needed, which can adapt, extract and give additional information to the AUI. This paper focuses mainly on this navigation module, not yet fully available in the literature, in a way it can be integrated in the AUI. Also extremely important is that this "emergent" technology is tested using Technological Acceptance Models (TAM), by using for instance Unified Theory of Acceptance and Use of Technology (UTAUT2) [45, 58].

3 Adaptive navigation

This section presents the navigation problem and proposes a method to build adaptive solutions according to the users' capacities and preferences. The problem is presented using a simulated visit to a museum, and then, the same methodology is applied to the city of Faro (real streets and POIs).

3.1 Preliminaries and problem definition

Let N_0 be a network that represents a space to be visited. For instance, Fig. 1, left, presents a museum where each node is a POI and on the right is sketched a network of the city of Faro (Portugal). In both cases, the color nodes are POIs (blue nodes) and the white ones are edge intersections, called auxiliary nodes. The presence of those auxiliary nodes is the main difference between the two presented networks. Each edge of the network is associated with a traversing time—the time to go from one node to the adjacent one. On the other hand, the POIs have a predetermined visiting time, which will be considered if the visitor is to explore that POI, and auxiliary nodes have visiting time equal to zero.

Depending on the type of network, each POI is categorized according to some classification system which also depends on the type of space and the type of users that use the application. For instance, a POI in a city, like Faro, can be categorized as juvenile, shopping, science, museum, church, theater, monument, kids park, edification, sight-seeing, etc. In the case of a geographical region, a more complete classification system can be derived from the GeoNames geographical database, where each feature is categorized into one out of nine feature classes and further sub-categorized into one out of 645 feature codes [12, 24]. If a museum (or similar) is considered, then other classification systems are adaptable to our system as, for instance, Iconclass [28], which is a hierarchically ordered collection of definitions of objects, people, events and abstract ideas that serve as the subject of an image. Art historians, researchers and curators use it to describe, classify

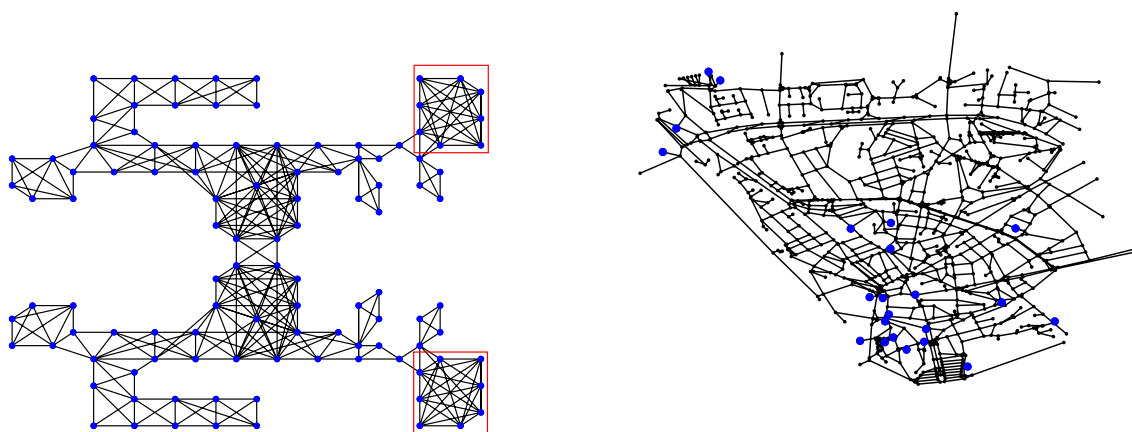


Fig. 1 Examples of networks: a museum where every node is a POI on the left and the map of the city of Faro (Portugal) having POIs (blue nodes) and auxiliary nodes (color figure online)

and examine the subject of images represented in various media such as paintings, drawings and photographs [13, 28, 29]. Also the Art & Architecture Thesaurus (AAT), the Getty Thesaurus of Geographic Names (TGN) and the Union List of Artist Names (ULAN) are structured vocabularies that can be used to improve the access to information about art, architecture, and material culture. The Cultural Objects Name Authority (CONA) is currently in development. It compiles titles, attributions, depicted subjects and other metadata about works of art, architecture and cultural heritage, both extant and historical; metadata is gathered or linked from museum collections, special collections, archives, libraries, scholarly research and other sources [2, 25]. Although not used at this point, the above classification systems can easily be adapted and included in the proposed work, as seen next.

The overall system can be designed to enhance accessibility and fight info-exclusion, as the previous POI classification systems can be extended to include features which reflect the degree of impairment of the visitor. For instance, painting POIs are probably not adequate for a blind person, and therefore, their classification relative to blind people would be very low. Furthermore, the network can also be designed with the impairments in mind, including information of the edges and POIs that are possible to be used by the visitors (e.g., stairs will not be included in the network edges if the user uses a wheelchair or has some other mobility difficulties).

In short, the network is a structure $N_0 = (V_0, E_0, d_0, t, C)$ where V_0 is the set of nodes which can be POIs or auxiliary nodes, $E_0 \subset V_0 \times V_0$ is the set of edges (each one connecting two nodes), $d_0 : E_0 \rightarrow \mathbb{R}_0^+$ is a function that associates with each edges its traversing time, $t : V_0 \rightarrow \mathbb{R}_0^+$ associates with each node the expected visit time, and

since each POI can be categorized in more than one class, $C : V_0 \rightarrow \{0, 1, \dots, 5\}^m$ classifies each node according to m classes in a scale of 0 to 5.

Given a maximum total visit time (T) and a vector of user's preferences ($UP \in \{0, 1, \dots, 5\}^m$), with each component associated with each classification class, the main problem is to discover an optimal walk $R = (n_s, n_1, n_2, \dots, n_k, n_t)$, where n_s is the starting node, n_t is the ending node and $n_i (i = 1, 2, \dots, k)$ are the POIs to be visited. Just recall, a walk is defined as any route through a network, from node to node along edges, which can end on the same node on which it began or on a different node, and can travel over any edge and any node any number of times. To avoid unnecessary computation, the algorithms will use a pre-computed network $N = (V, E, d, t, C)$ derived from N_0 , where V is the set of POIs and possible start and end nodes, $E \subset V \times V$ is the set of shortest paths between the nodes in V (shortest paths computed over N_0) and $d : E \rightarrow \mathbb{R}_0^+$ is the length of those shortest paths.

The problem is intrinsically multi-objective [16, 37] where several goals can be optimized, such as the total walked distance, the number of POIs visited, the total visit time, the time spent while observing POIs, and the validity of the walk in terms of observing the users' preferences. Given a walk $R = (n_s, n_1, n_2, \dots, n_k, n_t)$, this work addresses three objectives which are to be minimized, namely:

- The user's preferences cost given by

$$W_1(R) = 1 - \frac{\sum_{p \in R - \{n_s, n_t\}} \lambda(p)}{|R| - 2}, \quad (1)$$

where

$$\lambda(p) = \frac{\sum_{i=1}^m C_i(p)^{UP_i}}{\sum_{i=1}^m 5^{UP_i}}, \tag{2}$$

C_i is the classification value for the i th category and UP is a vector of user’s preferences. If a POI satisfies the users’ preferences, then λ will be approximately equal to 1. If all POIs in the visit satisfy the users’ preferences, then $(\sum_{p \in R - \{n_s, n_t\}} \lambda(p)) / (|R| - 2)$ will also be approximately equal to 1, and therefore, $W_1(R)$ will be approximately equal to 0.

- The *time spent observing POIs cost* is given by

$$W_2(R) = 1 - \frac{\text{time visiting POIs}}{\text{total visit time}}, \tag{3}$$

where the time spent visiting POIs is given by

$$\sum_{i=1}^{|R|-1} t(R_i)$$

and the total visit time is the time spent visiting POIs plus the time to go from POI to POI, i.e.,

$$\sum_{i=1}^{|R|-1} t(R_i) + \sum_{i=0}^{|R|-1} d(R_i, R_{i+1}). \tag{4}$$

If the time spent walking from POI to POI is low, then (time visiting POI)/(total visit time) is approximately 1, and therefore, $W_2(R)$ is approximately 0.

- The *diversity cost* is given by

$$W_3(R) = 1 - \frac{|R| - 2}{\text{total number of POIs}}. \tag{5}$$

This objective is related to the percentage of POIs visited. A larger number of visits will return $W_3(R)$ near 0. The solution of a multi-objective problem is a set of trade-off solutions called Pareto set [16, 37]. In the Pareto (or efficiency) order relation, a solution R is said to dominate another solution S , $R < S$, when R is not worse than S for all objectives and there is at least one on which it is strictly better, i.e., considering the 3 objectives (W_1, W_2 and W_3),

$$R < S \Leftrightarrow \begin{cases} \forall_{i \in \{1,2,3\}} : W_i(R) \leq W_i(S) \\ \exists_{i \in \{1,2,3\}} : W_i(R) < W_i(S). \end{cases} \tag{6}$$

A single “optimal” walk can be obtained by computing the entire Pareto set and then selecting an element from that set. However, the computation of the Pareto set is in general extremely expensive which implies that the end user might be satisfied with an approximation to the Pareto set and in particular he/she can be pleased with a single solution that observes its interests.

A simpler way to compute a single “optimal” can be achieved by transforming the original multi-objective problem into a single-objective problem by, for instance, redefining the objective function as a weighted sum function. The solution obtained with the weighted sum method is known to be Pareto optimal [37]. In our case, a slightly different (single objective) weighted function was designed as

$$F(R) = \omega_1 \times W_1(R) + \omega_2 \times \frac{W_2(R) + W_3(R)}{2}, \tag{7}$$

where $\omega_1, \omega_2 \in [0, 1]$ are weights that can be used to give more importance to one of the objectives and $\omega_1 + \omega_2 = 1$. Notice that cost W_2 and W_3 are correlated and therefore were associated in a single summand. This simplification of the formula allows to use less combinations of the weights (using only two, ω_1, ω_2) while not significantly affecting the results.

3.2 Algorithmical approaches

This section explains the algorithmic approach used to design the walks. At this stage, an ant colony optimization (ACO) algorithm [17] was selected. ACO algorithms are meta-heuristics based on the collective behavior of the majority of the ant colonies, where sets of agents compute new solutions based on artificial pheromone trails left by the previous agents. Technically, those pheromone trails are numerical values reflecting the best solutions found so far. ACO algorithms have a background of success solving many multiple objective optimization problems [6, 22, 38]. The general process can be described as follows. During a set of cycles, a collection of solutions based on the pheromone matrices and possible heuristics are computed. These solutions are then evaluated and used to update the pheromone matrices for the next cycle. The overall procedure is supported by the positive and negative feedback generated by pheromone updating strategies. Algorithm 1 sketches a general ACO.

The process described in Algorithm 1 is common to the majority of the ACO implemented solutions, varying mainly in step 5. Our approach includes two methods to compute a solutions as explained next.

Algorithm 1 Ant Colony Optimization Algorithm.

Ensure: : (Approximation) to the optimum solution
 1: Set parameters
 2: Set pheromone trail, $\tau = [1]$
 3: **repeat**
 4: **for all** ants **do**
 5: Build a solution using pheromone trails and heuristics
 6: Apply local search to the solution {Optional}
 7: **end for**
 8: Update the pheromone trail using the solutions obtained in step 5
 9: **until** stopping criteria is met
 10: **return** best achieved solution

3.2.1 Method A

The first method—Method A—considers a starting (n_s) and an ending (n_t) node to define an initial walk, $R = [n_s, n_t]$. In the next step, for each non-visited POI p the best position k (in terms of walking time) in R is found, and the pair (p, k) is kept in a candidate set, CS , if the total time constraint T is not violated by pushing p into position k of R . Now, (a) if the candidate set is not empty, select the next node and position (p, k) to be placed in the walk (and push it into R) according to the following formula

$$(p, k) = \begin{cases} \operatorname{argmax}_{(p,k) \in CS} [\tau(R_k, p)\tau(p, R_{k+1})]^\alpha \lambda(p)^\gamma & \text{if } q < q_0 \\ (p', k') & \text{if } q \geq q_0 \end{cases} \quad (8)$$

where:

- $\tau(x, y)$ is the amount of pheromone in the path $x \rightarrow y$;
- λ was defined in Eq. (2);
- α and γ are control parameters which allow to give more importance to the pheromone and/or preference factors. For instance, a large α will emphasize the use of the pheromone while a large γ will emphasize the users' preferences;
- (p', k') is a node and position pseudo-randomly selected from the candidate list using the probability function

$$P(p', k') = \frac{[\tau(R_{k'+1}, p')\tau(p', R_{k'+1})]^\alpha \lambda(p')^\gamma}{\sum_{(r,k) \in CS} [\tau(R_k, r)\tau(r, R_{k+1})]^\alpha \lambda(r)^\gamma} \quad (9)$$

After inserting the POI in the walk, reset the candidate set and repeat the previous steps. Otherwise, (b) if the candidate set was empty then walk R is returned, since there is no admissible insertion of a POI into R , and the method stops. Algorithm 2 outlines the described process.

Algorithm 2 Solution computation – Method A.

Require: : n_s (starting node), n_t (ending node), maximum allowed visit time (T), α, γ, q_0 , set of POIs
Ensure: : A walk
 1: $R = (n_s, n_t)$ {initial walk}
 2: $T_R = d(n_s, n_t)$ {initial traversing time}
 3: **while** True **do**
 4: $CS = \emptyset$ {Candidate set}
 5: **for all** not visited POIs, p **do**
 6: Find position, k , on the walk that minimizes the total traversing time if the POI, p , is to be inserted in that position, i.e., $k = \operatorname{argmin}_{i \in \{0, 1, \dots, |R|-1\}} T_R - d(R_i, R_{i+1}) + d(R_i, p) + d(p, R_{i+1})$
 7: **if** inserting p in R does not exceed the maximum visit time **then**
 8: $CS = CS \cup \{(p, k)\}$ {keep the candidate and the position}
 9: **end if**
 10: **end for**
 11: **if** $CS \neq \emptyset$ **then**
 12: Use Eq. (8) to choose a node p , and respective position k (obtained in step 6), between the candidates in CS .
 13: Push p into position k of R
 14: Update the traversing time, T_R
 15: **else**
 16: **return** R
 17: **end if**
 18: **end while**

3.2.2 Method B

Algorithm 3 Solution computation – Method B.

Require: : n_s (starting node), n_t (ending node), maximum allowed visit time (T), α, γ, q_0 , set of POIs
Ensure: : A walk
 1: $R = (n_s, n_t)$ {initial walk}
 2: $T_R = d(n_s, n_t)$ {initial traversing time}
 3: **while** True **do**
 4: $CS = \emptyset$ {Candidate set}
 5: **for all** not visited POIs, p **do**
 6: **for** $k = 0, 1, \dots, |R| - 1$ **do**
 7: **if** $T_R - d(R_k, R_{k+1}) + d(R_k, p) + d(p, R_{k+1}) < T$ **then**
 8: $CS = CS \cup \{(p, k)\}$ {Check if p can be placed at position k without exceeding the maximum visit time and update CS }
 9: **end if**
 10: **end for**
 11: **end for**
 12: **if** $CS \neq \emptyset$ **then**
 13: Use Eq. (10) to choose a node p , and respective position k (obtained in step 8), between the candidates in CS .
 14: Push p into position k of R
 15: Update the traversing time, T_R
 16: **else**
 17: **return** R
 18: **end if**
 19: **end while**

The second method has similarities with Method A. The method begins by defining an initial walk, $R = [n_s, n_t]$, given a starting (n_s) and an ending (n_t) node. Then, for each non-visited POI p , all admissible insertions position k in R are found, and the pairs (p, k) are kept in a candidate set, CS . (a) If the candidate set is not empty, then select the next node and position (p, k) to be placed in the walk (and push it into R) according to the following formula

$$(p, k) = \begin{cases} \operatorname{argmax}_{(p,k) \in CS} \frac{[\tau(R_k, p)\tau(p, R_{k+1})]^\alpha \lambda(p)^\gamma}{[d(R_k, p)d(p, R_{k+1})]^\beta} & \text{if } q < q_0 \\ (p', k') & \text{if } q \geq q_0 \end{cases} \tag{10}$$

where β is a control parameter which allows to emphasize an heuristic which favors the insertions of nodes closer to the nodes already present in the walk R . Furthermore, (p', k') is a node and position pseudo-randomly selected from the candidate list using the probability function

$$P(p', k') = \frac{[\tau(R_{k'+1}, p')\tau(p', R_{k'+1})]^\alpha \lambda(p')^\gamma}{[d(R_{k'+1}, p')d(p', R_{k'+1})]^\beta} \cdot \frac{1}{\sum_{(r,k) \in CS} \frac{[\tau(R_k, r)\tau(r, R_{k+1})]^\alpha \lambda(r)^\gamma}{[d(R_k, r)d(r, R_{k+1})]^\beta}} \tag{11}$$

The remaining parameters were already introduced after Eq. (8). Now, as in the previous method, after inserting the POI in the walk, reset the candidate set and repeat the previous steps. Otherwise, (b) if the candidate set was empty then walk R is returned, since there is no admissible insertion of a POI into R , and the method stops. Algorithm 3 outlines the described process.

3.2.3 Pheromone update

The pheromone represents a central role in any ACO algorithm. Used in the building of the solutions, the pheromone trail is updated after each cycle according to formula

$$\tau(e) = \rho\tau(e) + \Delta(e), e \in E, \tag{12}$$

where (1) $\tau(e)$ is the pheromone associated with path e ; (2) $\rho \in [0, 1]$ is called the persistence factor ($1 - \rho$ is the evaporation factor). The smaller the values of ρ are, the smaller quantity of information, used in one cycle, is transmitted to following cycle; (3) $\Delta(e)$ is the pheromone reinforcement associated with path e and is computed using the formula

$$\Delta(e) = \sum_{R \in S_e} \frac{Q}{F(R)}, \tag{13}$$

where S_e are the computed solutions containing path e and Q is a value with the same magnitude of the solutions.

4 Computational results

A set of tests were run varying the parameters such that $\alpha, \beta, \gamma \in \{0, 1, 3\}$, $q_0 \in \{0, .75\}$, 2-OPT local optimizer [14] turned on and off, $\rho = .9$, and $\omega_1, \omega_2 \in \{0, .5, 1\}$ (such that $\omega_1 + \omega_2 = 1$). For each set of parameters, 25 runs were made with 25 cycles of 25 ants over the network present in Fig. 1, left. Each POI in the network was classified according to eleven categories. The network presents two areas, inside the red rectangles, which were categorized as highly adequate for senior people and also with high classifications in “sixties photography.” The remaining categories and POIs were classified randomly. The visit time was also generated randomly, except for two of the previously classified POIs (one in inside each red rectangle) which were defined having a large visit time.

Since it is impracticable to present all results, Table 1 resumes the best mean (and standard deviation) results for the W_1, W_2 and W_3 cost functions, presented in the definition of the problem (Sect. 3). Furthermore, besides the methods parameters ($\alpha, \beta, \gamma, q_0, \omega_1, \omega_2$ and 2-OPT on/off), the aggregated cost function (F), the number of visited POIs and time spent observing POIs (from 90 time units) are also presented. The last two values are shown since they are more “legible” values. Finally, please recall that all costs were to be minimized.

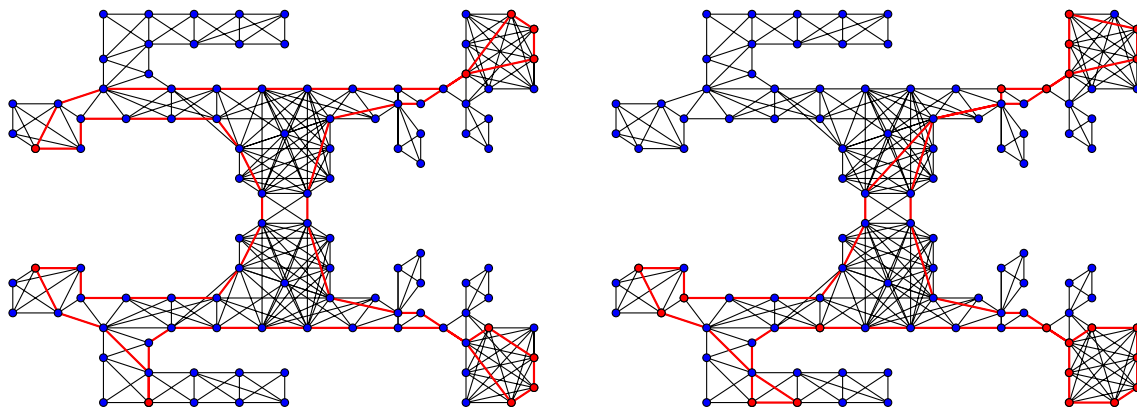
From Table 1, some conclusions can be drawn. The best result for W_1 shows solutions where the mean number of visited POIs is equal to 11 (Fig. 2 left shows a typical result for the best set of parameters). On the other hand, the mean number of visited POIs raises to over 30 if the best results for W_2 or W_3 are chosen (Fig. 2 right and Fig. 3 left show typical results for the best sets of parameters). Similarly, the visit time expended observing POIs is much smaller when considering the best results of W_1 (with a mean value around 39.3 of the 90 times units), against the best results of W_2 and W_3 (with a mean value around 70 of the 90 times units).

The best results for the W_1 cost were naturally obtained for $\omega_1 = 1$ and $\omega_2 = 0$. On the other hand, it was also natural that the best results for the W_2 and W_3 costs were obtained for $\omega_1 = 0$ and $\omega_2 = 1$. In this sense, Table 2 presents the best values for the aggregate cost function F when a balanced preference set was considered between the costs weights, i.e., $\omega_1 = \omega_2 = .5$. When compared with the previous results, the resulting walks are more balanced in the sense

¹ β is not used in Method A.

Table 1 Best results for costs W_1 , W_2 , and W_3 (mean and SD)

	Method	W_1	W_2	W_3	POIs Visit time	No. POIs	F	α	β	γ	q_0	2-OPT	ω_1	ω_2
Best results for W_1	B	.002 (.00)	.484 (.04)	.899 (.01)	39.3 (2.5)	11.0 (.8)	.002 (.00)	1	0	3	.75	Y	1	0
	B	.002 (.00)	.491 (.03)	.899 (.01)	39.1 (1.9)	10.9 (.7)	.002 (.00)	3	0	3	.75	Y	1	0
	B	.002 (.00)	.568 (.02)	.900 (.01)	38.8 (1.7)	10.8 (.8)	.002 (.00)	1	0	3	.75	N	1	0
	B	.002 (.00)	.568 (.02)	.899 (.01)	38.7 (1.8)	10.9 (.8)	.002 (.00)	1	0	1	.75	N	1	0
	B	.002 (.00)	.572 (.02)	.902 (.01)	38.4 (1.6)	10.6 (.6)	.002 (.00)	3	0	3	.75	N	1	0
	B	.002 (.00)	.440 (.02)	.907 (.00)	37.0 (.0)	10.0 (.0)	.002 (.00)	0	0	1	.75	Y	1	0
	B	.002 (.00)	.454 (.03)	.907 (.00)	36.8 (.4)	10.0 (.0)	.002 (.00)	0	0	3	.75	Y	1	0
Best results for W_2	B	.783 (.02)	.110 (.02)	.702 (.00)	75.0 (.7)	32.2 (.5)	.423 (.00)	1	3	0	.75	Y	0	1
	B	.823 (.04)	.114 (.03)	.720 (.01)	72.2 (1.3)	30.2 (.6)	.431 (.00)	0	1	0	.75	Y	0	1
	B	.822 (.03)	.116 (.03)	.706 (.01)	74.4 (1.1)	31.8 (.8)	.427 (.00)	1	1	0	0	Y	0	1
	B	.827 (.04)	.120 (.02)	.697 (.01)	76.2 (.8)	32.7 (.7)	.419 (.00)	1	3	0	0	Y	0	1
	B	.825 (.04)	.121 (.02)	.711 (.01)	71.7 (.8)	31.2 (.8)	.428 (.00)	0	3	0	0	Y	0	1
	B	.833 (.05)	.123 (.02)	.703 (.01)	76.1 (1.2)	32.1 (.7)	.421 (.00)	1	1	0	.75	Y	0	1
	B	.815 (.05)	.128 (.02)	.700 (.01)	76.4 (.6)	32.4 (.8)	.420 (.00)	3	3	0	0	Y	0	1
Best results for W_3	A	.757 (.02)	.173 (.01)	.688 (.01)	73.8 (1.0)	33.7 (.7)	.431 (.00)	3	-	0	0	N	0	1
	A	.764 (.04)	.166 (.01)	.690 (.01)	74.0 (1.2)	33.4 (1.2)	.431 (.00)	3	-	0	0	Y	0	1
	A	.770 (.03)	.167 (.01)	.692 (.01)	74.2 (1.1)	33.3 (1.0)	.430 (.00)	1	-	0	.75	Y	0	1
	A	.759 (.03)	.176 (.01)	.694 (.01)	73.9 (.9)	33.0 (1.0)	.434 (.00)	1	-	0	0	N	0	1
	A	.792 (.03)	.174 (.01)	.696 (.01)	73.9 (1.1)	32.9 (1.0)	.434 (.01)	1	-	0	.75	N	0	1
	B	.827 (.04)	.120 (.02)	.697 (.01)	76.2 (.8)	32.7 (.7)	.419 (.00)	1	3	0	0	Y	0	1
	A	.775 (.03)	.220 (.02)	.697 (.01)	69.6 (1.5)	32.8 (.9)	.459 (.01)	3	-	0	.75	Y	0	1

**Fig. 2** Example of a walk in a museum for the best results for W_1 (left) and for W_2 (right)

that the mean number of visited POIs is around 20 and the POIs visit time around 60 time units of the maximum 90 allowed. Figure 3 right shows a typical result for the best set of parameters.

As final observations, the 2-OPT local optimizer presents an important role as the majority of the best solutions were obtained when it was active. Tables 1 and 2 also show that Method B appears with more frequency in the best results, although the difference in terms of costs was not expressive. Nevertheless, Method B is computationally more demanding than Method A which might pose a doubt as to which

method to use in a real-time application accessed multiples times simultaneously.

The same algorithms were applied to the city of Faro, using the best parameters for F and $\omega_1 = \omega_2 = .5$, i.e., $\alpha = \beta = 1$, $\gamma = 3$, $q_0 = 0$ and 2-OPT activated. The network has 24 POIs, classified between 1 and 5 in 11 categories, e.g., “shopping,” “museum,” “church,” “theater” or “monument.” The visitor was characterized as looking for “churches” (church preference was set to 5 and the others between 0 and 2) and having 1500 time units to spend. Figure 4 shows the results for a single run of each

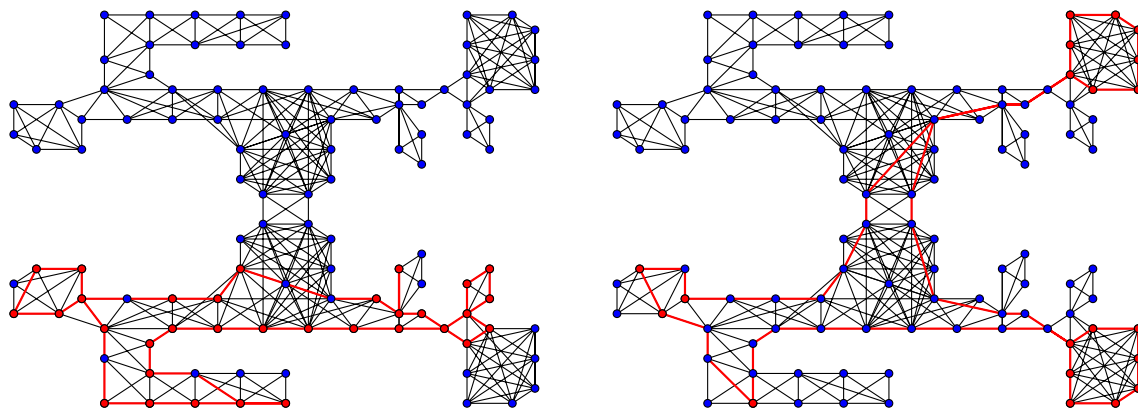


Fig. 3 Example of a walk in a museum for the best results for W_3 (left) and for F when $\omega_1 = \omega_2 = .5$ (right)

Table 2 Best results for costs F when $\omega_1 = \omega_2 = .5$ (mean and standard deviation)

Methods	W_1	W_2	W_3	POIs visit time	No. POIs	F	α	β	γ	q_0	2-OPT	ω_1	ω_2
B	.019 (.02)	.234 (.05)	.822 (.01)	62.6 (1.9)	19.2 (.9)	.283 (.01)	1	1	3	0	Y	.5	.5
B	.013 (.01)	.253 (.03)	.825 (.01)	61.5 (2.3)	18.9 (.9)	.283 (.01)	1	0	3	.75	Y	.5	.5
B	.030 (.01)	.238 (.05)	.817 (.00)	64.2 (.9)	19.8 (.5)	.285 (.00)	1	1	3	.75	Y	.5	.5
A	.023 (.01)	.278 (.02)	.817 (.00)	64.8 (2.1)	19.7 (.4)	.285 (.00)	1	–	3	0	Y	.5	.5
A	.021 (.01)	.280 (.02)	.819 (.01)	64.7 (2.2)	19.6 (.6)	.285 (.00)	1	–	3	0	N	.5	.5
A	.014 (.01)	.292 (.02)	.820 (.00)	63.6 (2.0)	19.4 (.5)	.285 (.00)	1	–	3	.75	N	.5	.5
A	.003 (.00)	.316 (.00)	.824 (.00)	61.2 (.2)	19.0 (.0)	.286 (.00)	0	–	3	.75	Y	.5	.5

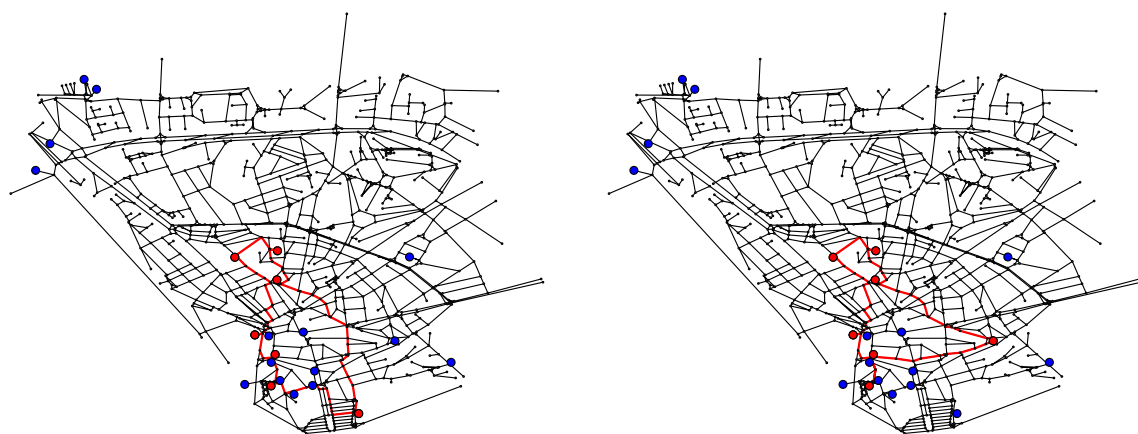


Fig. 4 Example of walks in the city of Faro for the best parameters (for F and $\omega_1 = \omega_2 = .5$): Method A on left and Method B on right

of the methods: Method A on left and Method B on right. The results for both methods are similar with Method A obtaining $W_1 = .040$, $W_2 = .417$, $W_3 = .636$, and $F = .284$, and Method B returned $W_1 = .040$, $W_2 = .409$, $W_3 = .636$ and $F = .281$. In both cases, the solutions start in the same node and passes through six POIs classified as churches.

5 Adaptive user interface

Having the adaptive navigation module prepared, its integration with the AUI is an important step. A full AUI, in its limit, has different layouts and contents for each of

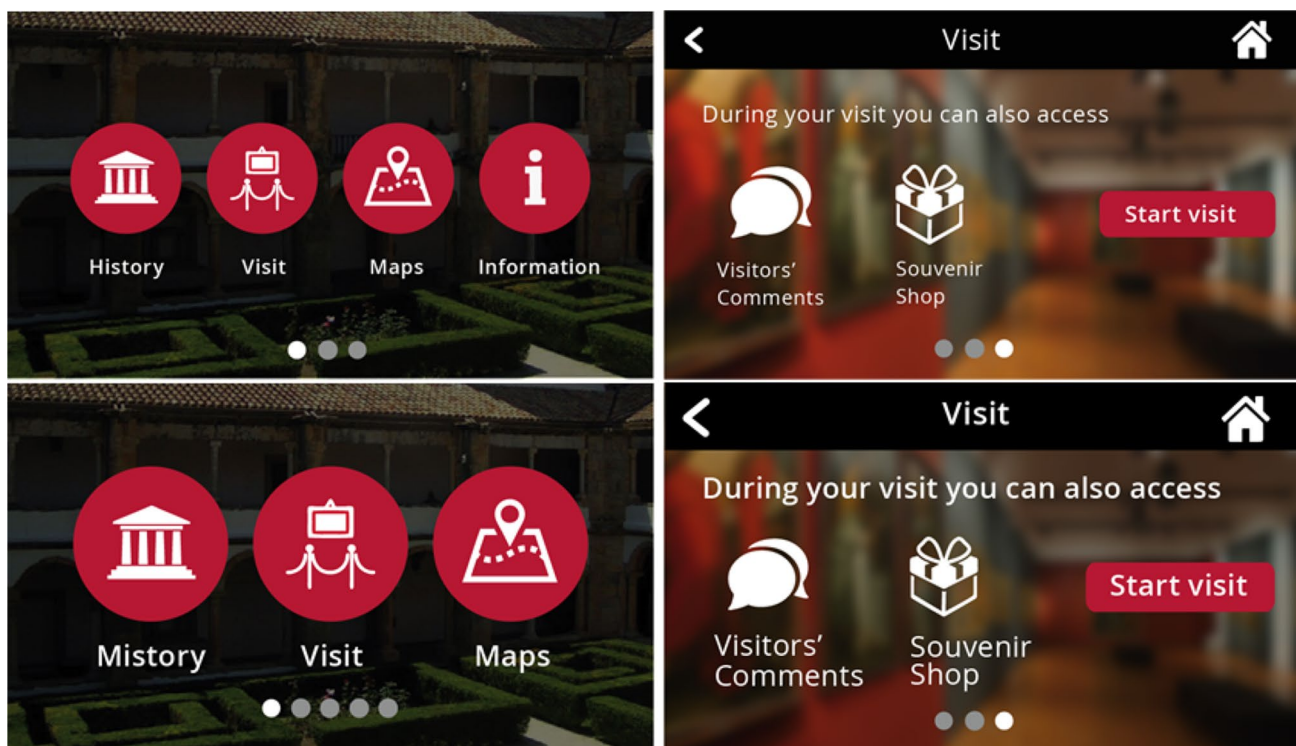


Fig. 5 Top row, distinct mock-up for the general visitors (left) and elder people (right). Bottom row, navigation interfaces: menu to launch the map (left), map pinpointing the users's position (right)

the user's views. Nevertheless, given a large enough set of people, partial or complete layout can be shared by different users. The same layout and structure can also be used in multiple views, e.g., when showing information about different museums' objects to the same user, it might be natural to maintain the same layout. In this case, for objects in the same category, the displayed contents (e.g., figures, text, or videos) are inserted in a same layout. Maintaining the layout helps the user to feel familiar with the application, avoiding the need to keep learning where the contents are shown, or which and where are the actions he/she will be able to perform [46].

Even when the layout is the same, for different users the content could be different, adapting to its needs. For instance, a kid that knows how to read probably should not have the same layout and/or contents as a kid that does not know how to read; alternatively, an expert might have distinct details about the objects compared to a "normal user"; Moreover, if the user is an expert but has vision problems the contents might be equal/similar but the display could be adapted by, for instance, using larger fonts, more contrast or larger buttons. Figure 5 shows a possibility for a visually simple and strong graphic language mock-up. The first row presents two views for an average user App version and the bottom row two views for senior version where higher font size and weight variations,

bigger buttons and slightly higher contrast were applied. The discussion of features that should be used to implement this kind of App for senior citizens was presented by Rodrigues et al. [45].

With the principle of on-the-fly adapting UI, in order to optimize the App's memory and CPU utilization, each layout (or partial layout) could be build only once, kept in memory, and called every time it is required. It is important to stress that this methodology can easily be achieved using the Unity [53] development platform. In this sense, each view seen by the users was separated in its structure/layouts and its contents, which allows to place (distinct) contents in the (same) layout at different execution points. For the detailed implementation of the adaptive card design UI, see [46]. Both the layouts and the contents are then stored in a relational database. When the App requests the necessary data for a certain museum from the web servers, a web service will execute proper queries to the database, obtaining the layouts and contents which are then sent as JSON documents, and decoded in the App to build the views.

Another advantage of the AUI, beyond the on-the-fly adaptation to the users, is the facility to adjust too different designer and testing proposals. Figure 6 shows a different design example of the museum's App views, now based on cards. Finally, Fig. 7 presents several views of the App's present version.

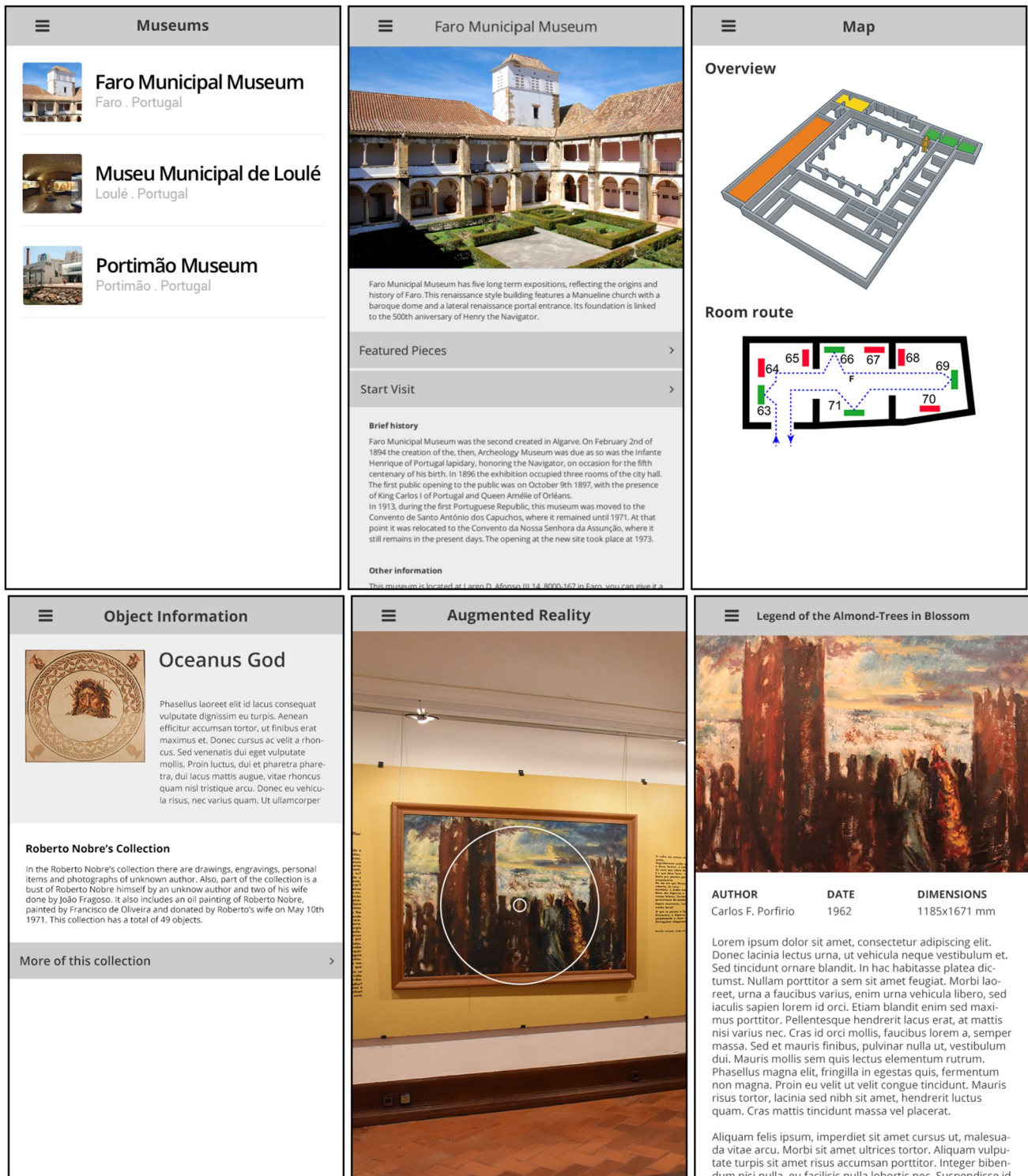


Fig. 6 Top to bottom, left to right: list of available museums, specific museum information, museum map with a route calculated, example of card piece, image recognition (AR), and information about the detected piece

Besides the storage of the layouts and contents, the database represents a fundamental component of the system, where (authorized) user information and specifications are kept. This information allows the use of machine learning

techniques to harvest relevant data used to build the views and, at the same time, select the conditions for the adaptive navigation module to generate the user’s “route.” For

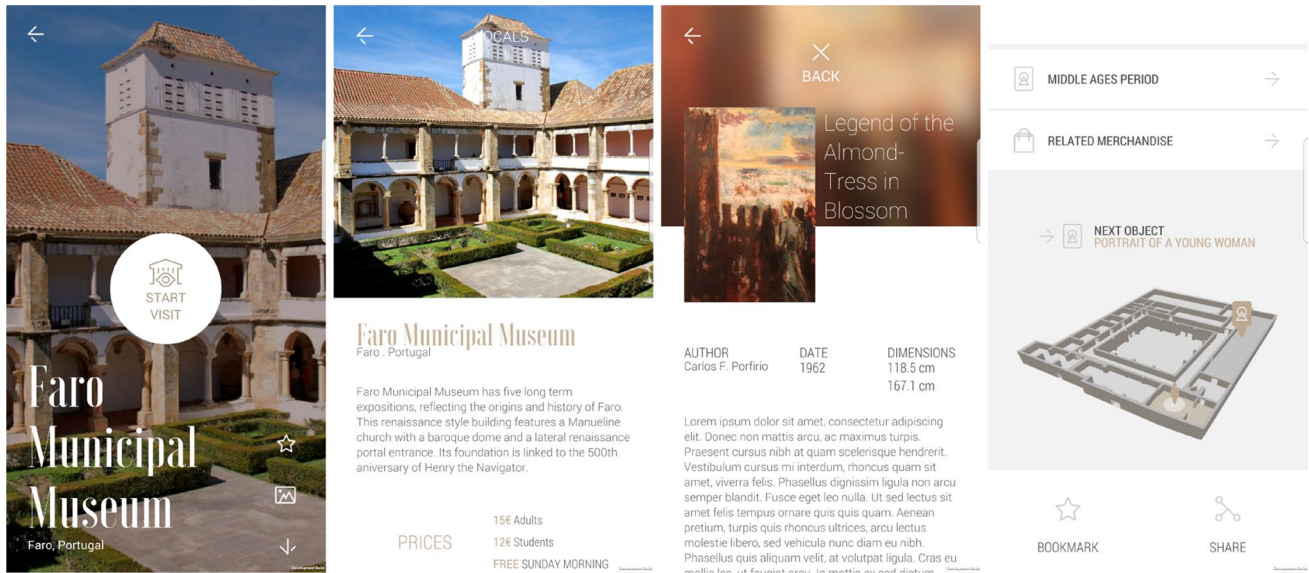


Fig. 7 Examples of views from the present version of the App

instance, that data are used to suggest museum's objects or souvenirs to be bought in the museum's store [7].

Returning to navigation, with a solution proposed in Sect. 3, the route suggested to the user is build using an optimization procedure which pinpoints a set of objects from the museum's collection. The problem now resumes to the display of the computed route in a meaningful manner to the user. A challenge is to locate the user in the museum. Since most museums exhibitions are located inside buildings, generally implying the impossibility of using Global Positioning System (GPS), whenever needed, it was decided to use a hybrid solution supported on "AR localization" and beacons [18, 40]. Those beacons are portable battery-powered devices that use the Bluetooth Low Energy (BLE) radio protocol and can be easily distributed all over the building at low cost (typically one per museum room). The user's distance to the beacons can be estimated, used to do a triangulation and predict the user's location, usually restricting him/her to a "small" area inside an exhibition room. The above information is integrated with the precise localization of the user every time he/she uses the AR, i.e., when the user points the mobile device to a museum piece, and the piece is recognized to trigger the AR, beyond the extra contents that the App gives to the user, his/her exact position related to the detected piece is send to the navigation module. Although the computer vision module, used to detect the pieces and the AR deployment, is out of the paper's scope, detailed information about that module can be found in [41].

The integration of the routing system in the App was done as follows. The App's menu allows to launch the map's features among other things, such as executing a virtual visit to the collections and run functions to plan the user's visit.

After launching the maps view, the user's position is pinpointed in the map allowing him/her to have an idea of where the objects suggested for its visit are (Fig. 6, top right). The App enters then in AR navigation mode, where computer vision is used to detect the objects to be visited, marks them in the screen and suggests a path to them (Fig. 6, bottom middle and top right, with the contents proposed by the AR in the bottom right).

6 System architecture

Given the previous sections, the relevant part of the overall system architecture is now depicted in Fig. 8. The user's mobile device serves three main purposes: display information, display forms to configure/get the user's preferences, and get and send the user's location to the data center. The first two purposes are implemented using the AUI methodology (see Sect. 5). As already mentioned, also in Sect. 5, the localization of the user in indoor places is attained by combining "AR localization" and beacons. The user's location is approximated in the room he/she is visiting, but allows to pinpoint the user's position in the interface and guide him/her in the planned route. The information between the mobile device and the data center is passed through the available wireless channels (e.g., Wi-fi or mobile network).

The data center includes the necessary storage and computational hardware and software. Presently, information is stored in a relational database which includes user's data (personal data, preferences, etc.), interface cards data (see Sect. 5), contents for the cards (e.g., museum's information or pieces information, see Sect. 5), museum's maps,

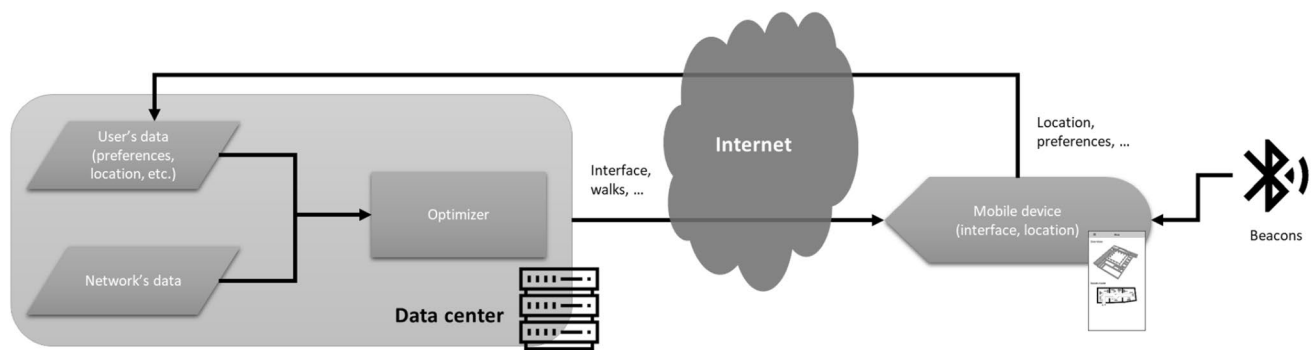


Fig. 8 Diagram of the systems' architecture

etc. Whenever required by the users (e.g., by accessing the mobile application), the AUI cards are built and sent, along with the contents, to the mobile device. This allows to have distinct interfaces for the different users and museums, but also allows to deploy updates to the interfaces and contents on the fly, without the need of updating the mobile device application via stores.

Communications between the internal data center systems and data center and mobile device (both ways) are made through RESTful web services allowing to develop the main features as separated modules. Furthermore, this modular approach will allow easy upgrades and updates to the modules' technologies, as long as the web services's protocols are observed.

When the user accesses the route planner card, the route optimizer is called in the data center. The optimizer takes as input user preferences, location and the museum network (see Sect. 3) and returns a suggested route in the museum. Prepared routes consider the user at the museum entrance. The result is then send and displayed in the mobile device.

7 Discussion and conclusion

The way people experience cultural heritage is changing. Traditional visits where everyone, despite their interests or limitations, have to follow a predetermined route are no more the best way to do it. Furthermore, many times the number of points of interest is also large and the time available to explore them is limited, making impossible to experience all of them. Also important is to enhance accessibility and fighting info-exclusion by giving the needed users tools to plan their visit and to access data in a proper way.

This paper proposed the integration of a mobile adaptive augmented reality navigation system for museums. The navigation problem was formulated in order to compute optimal paths inside a network containing sets of POIs. Two methods, supported on the ACO algorithms to build, in near real time, walks which translate the user's preferences and

limitations, were explored. In this sense, both the network and the users' preferences have to be designed with the same insight. The results shown good solutions which translate the data and the preferences. Those results were then integrated in the mobile application in order to guide the users through the museum's rooms.

As future work, naturally, many things can be further developed and tested. For instance, it can be useful to carry out exhaustive stress tests like larger networks, larger number of categories, implementation of the demanding methods in computational devices with less capacity (like mobile devices), etc. The features aspect can also be further explored. One idea is to adapt the walk as the users navigate the network. For instance, the user might spend more time near a particular piece, or simply make a pause, which, given the limited visit time, must be reflected in the proposed walk. The walk should also adapt to the users' way of exploring the space. If the users spend much time near a piece which is classified "outside" the initial preferences, maybe the application should suggest other pieces of that type. Other features include the constructions of the walks based on the expected "occupation" of the POIs, reservation of time slots to the more wanted POIs or past information collected from users with similar interests.

Alternatives to the proposed solution include the use of recommendation systems combined with path optimizers. The recommender systems, part of the machine learning research field, use mathematical and computer science techniques to find patterns and correlations in order to build models [35, 44]. Those recommender systems are in general classified as content based, when features associated with the items characterize them, or collaborative filtering when supported in the items vs. users relations [27]. In both cases, the objective is to suggest items to the users. The path optimization can then be done using well-known deterministic or heuristic algorithms. The possible change would allow a modularization of the system (recommender system and path optimizer) and facilitate some aspects, as the need that each user has to inexhaustibly define its preferences.

Last but not least, the application is in the final stages of development and now in conditions of being tested with real users in a real museum environment, and validated (out of the scope of this paper) in terms of acceptance and use of new technology. One of the most influential models in the field of information systems to evaluate the acceptance and use of technology is the TAM, proposed by Davis [15]. However, TAM does not include factors that permit to investigate user's intentions when facing new technology, as considered by the UTAUT model [57], or constructs to investigate the consumer's use context, as considered by the UTAUT2 model [58], being an extension of UTAUT. In the context of the M5SAR project, the UTAUT2 model was considered as the appropriate model to study and validate the acceptance of new technologies and assess the performance level of the IUI/AUI development, simultaneously investigating the "perception of being easy to use" and the "perception of usefulness" in the consumer's context. Among other things, this will allow to measure the contributions to the fulfillment of the touristic experience and the satisfaction of users with specific characteristics and needs, once it integrates several constructs (e.g., hedonic motivation), while permitting to analyze the individuals' differences through age, gender and other variables.

Acknowledgements We thank our project leader SPIC - Creative Solutions [www.spic.pt] and the Museu Municipal de Faro.

References

- Akiki, P.A., Bandara, A.K., Yu, Y.: Adaptive model-driven user interface development systems. *ACM Comput. Surv.* **47**(1), 9:1–9:33 (2014). <https://doi.org/10.1145/2597999>
- Baca, M., Gill, M.: Encoding multilingual knowledge systems in the digital age: the Getty vocabularies. *Knowl. Organ.* **42**(4), 232–243 (2015)
- Balance: The balance: fine art museum apps for smart phones (2017). <https://goo.gl/WEsQRN>. Retrieved 13 Sept 2017
- Benouaret, I., Lenne, D.: Combining Semantic and Collaborative Recommendations to Generate Personalized Museum Tours, pp. 477–487. Springer, Berlin (2015). https://doi.org/10.1007/978-3-319-23201-0_48
- Candlin, F.: The dubious inheritance of touch: art history and museum access. *J. Vis. Cult.* **5**(2), 137–154 (2006). <https://doi.org/10.1177/1470412906066906>
- Cardoso, P., Jesus, M., Márquez, A.: ϵ - DANTE : an ant colony oriented depth search procedure. *Soft Comput.* **15**(1), 149–182 (2011). <https://doi.org/10.1007/s00500-010-0543-9>
- Cardoso, P.J.S., Rodrigues, J.M.F., Pereira, J.A.R., Sardo, J.D.P.: An Object Visit Recommender Supported in Multiple Visitors and Museums, pp. 301–312. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-58706-6_24
- CEUD: Centre for excellence in universal design (2016). <http://goo.gl/fUHPj6>. Retrieved 13 Sept 2017
- CHES: CHES - cultural heritage experiences through socio-personal interactions and storytelling (2017). <http://www.chesexperience.eu/>. Retrieved 13 Sept 2017
- CM: Multitouch wall Cleveland Museum (2013). <https://goo.gl/kI2oCh>. Retrieved 13 Sept 2017
- Conati, C., Carenini, G., Toket, D., Lallé, S.: Towards user-adaptive information visualization. In: *AAAI*, pp. 4100–4106 (2015)
- Coughlan, T., Carletti, L., Giannachi, G., Benford, S., McAuley, D., Price, D., Locatelli, C., Sinker, R., Stack, J.: Artmaps: Interpreting the spatial footprints of artworks. In: *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, ACM, New York, NY, USA, CHI '15, pp. 407–416 (2015). <https://doi.org/10.1145/2702123.2702281>
- Coupré, L.D.: Iconclass, a device for the iconographical analysis of art objects. *Mus. Int. (Ed. Fr.)* **30**(3–4), 194–198 (1978)
- Croes, G.A.: A method for solving traveling-salesman problems. *Oper. Res.* **6**(6), 791–812 (1958)
- Davis, F.D.: Perceived usefulness, perceived ease of use, and user acceptance of information technology. *MIS Q.* **13**(3), 319–340 (1989). <https://doi.org/10.2307/249008>
- Deb, K.: *Multi-objective Optimization Using Evolutionary Algorithms*. Wiley, New York (2001)
- Dorigo, M., Stützle, T.: *Ant Colony Optimization*. MIT Press, Cambridge (2004)
- Faragher, R., Harle, R.: Location fingerprinting with bluetooth low energy beacons. *IEEE J. Sel. Areas Commun.* **33**(11), 2418–2428 (2015)
- Gajos, K., Weld, D.S.: Supple: automatically generating user interfaces. In: *Proceedings of the International Conference on Intelligent User Interfaces*, pp. 93–100. ACM (2004)
- Gajos, K.Z., Wobbrock, J.O., Weld, D.S.: Improving the performance of motor-impaired users with automatically-generated, ability-based interfaces. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 1257–1266. ACM (2008)
- Garcia, I., Sebastia, L., Onaindia, E.: On the design of individual and group recommender systems for tourism. *Expert Syst. Appl.* **38**(6), 7683–7692 (2011). <https://doi.org/10.1016/j.eswa.2010.12.143>
- García-Martínez, C., Cordon, O., Herrera, F.: An empirical analysis of multiple objective ant colony optimization algorithms for the bi-criteria TSP. In: *ANTS Workshop, Lecture Notes in Computer Science*, vol. 3172, pp. 61–72 (2004)
- Gavalas, D., Konstantopoulos, C., Mastakas, K., Pantziou, G.: Mobile recommender systems in tourism. *J. Netw. Comput. Appl.* **39**, 319–333 (2014). <https://doi.org/10.1016/j.jnca.2013.04.006>
- GeoNames: GeoNames (2017). Retrieved 13 Sept 2017 <http://www.geonames.org/>
- Getty: Getty (2017). Retrieved 13 Sept 2017 <http://www.getty.edu/>
- Gnome: User interface guidelines for supporting accessibility (2016). Retrieved 13 Sept 2017 <https://goo.gl/xGAH6f>
- Hu, Y., Koren, Y., Volinsky, C.: Collaborative filtering for implicit feedback datasets. In: *Eighth IEEE International Conference on Data Mining*, 2008. ICDM'08, pp. 263–272. IEEE (2008)
- Iconclass: Iconclass (2017). Retrieved 13 Sept 2017 <http://www.iconclass.nl>
- Isemann, D., Ahmad, K.: Ontological access to images of fine art. *J. Comput. Cult. Herit.* **7**(1), 3:1–3:25 (2014). 10.1145/2538030
- IW: InformationWeek: 10 fantastic iphone, android apps for museum visits (2015). <https://goo.gl/rYnhm2>. Retrieved 13 Sept 2017
- Jung, T., Chung, N., Leue, M.C.: The determinants of recommendations to use augmented reality technologies: the case of a korean theme park. *Tour. Manag.* **49**, 75–86 (2015)
- Karaman, S., Bagdanov, A.D., Landucci, L., D'Amico, G., Ferracani, A., Pezzatini, D., Del Bimbo, A.: Personalized multimedia content delivery on an interactive table by passive observation of museum visitors. *Multimed. Tools Appl.* **75**(7), 3787–3811 (2016)

33. Kralisch, A., Eisend, M., Berendt, B.: Impact of culture on website navigation behaviour. In: Proceedings of the HCI-International (2005)
34. Lindgaard, G., Fernandes, G., Dudek, C., Brown, J.: Attention web designers: you have 50 milliseconds to make a good first impression!. *Behav. Inf. Technol.* **25**(2), 115–126 (2006)
35. Lohr, S.: Netflix awards \$1 million prize and starts a new contest. *New York Times* 21 (2009)
36. Metallinou, A., Wollmer, M., Katsamanis, A., Eyben, F., Schuller, B., Narayanan, S.: Context-sensitive learning for enhanced audio-visual emotion classification. *IEEE Trans. Affect. Comput.* **3**(2), 184–198 (2012)
37. Miettinen, K.: *Nonlinear Multiobjective Optimization*. Kluwer Academic Publishers, Dordrecht (1999)
38. Mohan, B.C., Baskaran, R.: A survey: ant colony optimization based recent research and implementation on several engineering domain. *Expert Syst. Appl.* **39**(4), 4618–4627 (2012). <https://doi.org/10.1016/j.eswa.2011.09.076>
39. Morency, L.P., Mihalcea, R., Doshi, P.: Towards multimodal sentiment analysis: harvesting opinions from the web. In: Proceedings of the 13th International Conference on Multimodal Interfaces, pp. 169–176. ACM (2011)
40. Palumbo, F., Barsocchi, P., Chessa, S., Augusto, J.C.: A stigmergic approach to indoor localization using bluetooth low energy beacons. In: 2015 12th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), pp. 1–6. IEEE (2015)
41. Pereira, J., Sardo, J., Freitas, M., Veiga, R., Cardoso, P., Rodrigues, J.: Mirar: Mobile image recognition based augmented reality framework. In: International Congress on Engineering and Sustainability in the XXI Century (2017)
42. Qualcomm: Invisible museum (2016). <https://goo.gl/aSONKh>. Retrieved 13 Sept 2017
43. Reinecke, K., Bernstein, A.: Knowing what a user likes: a design science approach to interfaces that automatically adapt to culture. *MIS Q.* **37**(2), 427–453 (2013)
44. Ricci, F., Rokach, L., Shapira, B. (eds.): *Recommender Systems Handbook*. Springer, Berlin (2015)
45. Rodrigues, J., Lessa, J., Gregório, M., Ramos, C., Cardoso, P.: An initial framework for a museum application for senior citizens. In: Proceedings of the 7th International Conference on Software Development and Technologies for Enhancing Accessibility and Fighting Info-exclusion (2016)
46. Rodrigues, J.M.F., Pereira, J.A.R., Sardo, J.D.P., de Freitas, M.A.G., Cardoso, P.J.S., Gomes, M., Bica, P.: Adaptive Card Design UI Implementation for an Augmented Reality Museum Application. LNCS, vol. 10277, pp. 433–443. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-58706-6_35
47. Route Perfect: Route perfect (2017). <https://www.routeperfect.com/>. Retrieved 13 Sept 2017
48. Sardo, J., Semião, J., Monteiro, J., Pereira, J., Freitas, M., Rodrigues, J., Esteves, E.: Portable device for touch, taste and smell sensations in augmented reality experiences (Accepted for the International Congress on Engineering and Sustainability in the XXI cEntury – INCREaSE 2017) (2017)
49. Schuller, B.W.: Modelling user affect and sentiment in intelligent user interfaces: a tutorial overview. In: Proceedings of the 20th International Conference on Intelligent User Interfaces, pp. 443–446. ACM (2015)
50. SM: Science museum - atmosphere gallery (2016). <https://vimeo.com/20789653>. Retrieved 13 Sept 2017
51. Srbija: Srbija 1914 / augmented reality exhibition at historical museum of serbia (2015). <https://vimeo.com/126699550>. Retrieved 13 Sept 2017
52. Steichen, B., Conati, C., Carenini, G.: Inferring visualization task properties, user performance, and user cognitive abilities from eye gaze data. *ACM Trans. Interact. Intell. Syst.* **4**(2), 11 (2014)
53. Unity: Unity 3D (2017). <https://unity3d.com/pt>. Retrieved 13 Sept 2017
54. UVAM: Interactive ipad museum catalog (2016). <https://vimeo.com/31821923>. Retrieved 13 Sept 2017
55. Vainstein, N., Kuflik, T., Lanir, J.: Towards using mobile, head-worn displays in cultural heritage: user requirements and a research agenda. In: Proceedings of the 21st International Conference on Intelligent User Interfaces, pp. 327–331. ACM (2016)
56. van Hage, W.R., Stash, N., Wang, Y., Aroyo, L.: Finding Your Way Through the Rijksmuseum with an Adaptive Mobile Museum Guide, pp. 46–59. Springer, Berlin (2010). https://doi.org/10.1007/978-3-642-13486-9_4
57. Venkatesh, V., Morris, M., Davis, G., Davis, F.: User acceptance of information technology: toward a unified view. *MIS Q.* **27**(3), 425–478 (2003). <https://doi.org/10.2307/30036540>
58. Venkatesh, V., Thong, J., Xu, X.: Consumer acceptance and use of information technology: extending the unified theory of acceptance and use of technology. *MIS Q.* **36**(1), 157–178 (2012). <https://doi.org/10.2307/41410412>
59. Verbert, K., Manouselis, N., Ochoa, X., Wolpers, M., Drachsler, H., Bosnic, I., Duval, E.: Context-aware recommender systems for learning: a survey and future challenges. *IEEE Trans. Learn. Technol.* **5**(4), 318–335 (2012). <https://doi.org/10.1109/TLT.2012.11>
60. Wang, D., Xiang, Z.: *The New Landscape of Travel: A Comprehensive Analysis of Smartphone Apps*, pp. 308–319. Springer, Vienna (2012). https://doi.org/10.1007/978-3-7091-1142-0_27
61. WSJ: The Wall Street Journal: best apps for visiting museums (2015). <https://goo.gl/cPTyP9>. Retrieved 13 Sept 2017
62. Zhao, L., Lu, Y., Zhang, L., Chau, P.Y.: Assessing the effects of service quality and justice on customer satisfaction and the continuance intention of mobile value-added services: An empirical test of a multidimensional model. *Decis. Support Syst.* **52**(3), 645–656 (2012)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.