CAViAR: Context Aware Visual indoor Augmented Reality for a University Campus

Buti Al Delail, Luis Weruaga and M. Jamal Zemerly
Dept. Electrical and Computer Engineering
Khalifa University of Science, Technology and Research, Sharjah, UAE
{100037712, luis.weruaga, jamal}@kustar.ac.ae

Abstract—CAViAR is a mobile software system for indoor environments that provides to the mobile user equipped with a smartphone indoor localization, augmented reality (AR), visual interaction, and indoor navigation. These capabilities are possible with the availability of state of the art AR technologies. The mobile application includes additional features, such as indoor maps, shortest path, inertial navigation, places of interest, location sharing and voice-commanded search. CAViAR was tested in a University Campus as one of the technologies to be used later in an intelligent Campus environment.

Keywords—Context aware, indoor localization, visual computing, image detection, augmented reality.

I. INTRODUCTION

The outstanding computing capabilities in today’s high-end phones or “smartphones” in combination with its wealth of sensors, such as GPS (Global Positioning System), inertial sensors, camera, wireless receivers, are powering the mobile application sector to the extent of becoming the fastest growing one in information communication technologies. Knowledge about the orientation and location delivered by current handheld devices allows for context-aware information services tailored to the user. In that regard, the excellent performance in mobile visual computing [1] permits the detection and tracking of markers and planar objects in real time. This information can be used for mobile visual location recognition [2] in dense outdoor or urban areas where GPS measurements are either unavailable or inaccurate.

On the other hand, localization in indoor scenarios is mainly dominated by wireless-based positioning techniques in combination with inertial sensors, and others [3], [4]. The cooperation of computer vision and inertial measurements has been also explored in indoor environments [5], but this particular solution requires unfortunately uncomfortable “wearable” computing equipment. A breakthrough application that derives somewhat from mobile visual computing is augmented reality (AR), namely, the display of virtual 3D objects that merge seamlessly with the actual video scene captured by the device camera [6]. As AR is founded on detection of a known marker, as well as its tilt in 3D coordinates [7], when the marker is in a permanent static location, de facto localization is accomplished.

Following the previous concept, this paper proposes a scalable software architecture for a mobile application that provides indoor location detection and tracking based on a combination of image marker recognition and inertial measurements. This localization service is core to deliver a context-aware information system built around an augmented reality software layer. The AR layer informs the user of a close point of interest, overlaying self-explanatory 3D virtual objects related to the location on the real-time video capture. The points of interest nearby are also shown in a 360 degree fashion supported on the device compass readings. Any AR virtual object is “clickable” or “touchable”, such that the user obtains information related thereupon. CAViAR’s system was initially developed for the University campus where the authors work. Nevertheless, the idea can be extended to any scenario without major architectural changes. The structure of this paper follows: Section II presents the architecture of the global system, which includes a mobile application and the information Internet server; Section III describes with sufficient detail the main implementation aspects of each module; Section IV addresses the evaluation of the system; Finally, Section V brings the future prospects and conclusions.

II. SYSTEM OVERVIEW

The overview of the system architecture is shown in Fig. 1. The user localization process starts with the video capture from the device camera. The video frames are analyzed in real-time to detect and track well-known markers. These markers must correspond to high-contrast planar objects that are part of the landscape, and are thus assumed to be static. The image detection process is accomplished with Qualcomm’s Vuforia SDK [8], which delivers the distance and 3D orientation of the image marker (as in [7]). The target orientation and posture are further used by the AR layer to overlay 3D virtual objects that merge seamlessly with the scene in the video capture. Simultaneously, the location manager fetches from the local database the information related to the detected marker, such as its location, hence implicitly obtaining the current location of the mobile user. That location can be displayed on an indoor map. Indispensable in this process is the communication module, which
connects to the service server via a TCP socket, so insuring that the local database is up to date. Above this core layer, the AR 360-degree visualization of points of interest in the vicinity of the user is an important system feature. Other features, such as voice-commanded search of locations have been explored and implemented.

The system is scalable, and it can be extended to include multiple buildings, new locations and detectable markers without major architectural software modifications. In summary, the universal system would consist of different dynamic sub-systems whose input data, such as maps, coordinates, locations, markers, 3D objects, and contextual information would be stored in the server. It is thus clear that since the server provides the data, the application code need not be modified. The iPhone application has been published on Apple’s AppStore for members of the University campus, the process of data update being fully transparent to the user.

III. IMPLEMENTATION

The software implementation was modular, based on an object-oriented software architecture. The mobile App was implemented in the iPhone with the official iOS SDK (Software Development Kit). It could be implemented for other mobile devices (Vuforia SDK [8] is available for Android), but given this task would imply major software development efforts because of programming language disparity.

Regarding the programming languages used in the implementation, native iOS applications must be written in Objective-C in the most part. However, the common interface of the communication messages was written in C++ to provide conventional and portable messaging.

A. Mobile Application

1) Indoor maps: The indoor maps were implemented as an overlay on the Google maps. This technique allows the application to show multiple indoor maps of buildings. The map sets are downloaded from the server and stored in the application cache storage. Fig. 2 shows the University campus indoor maps overlaying Google maps.

2) Indoor Localization: The localization system combines several features including image-based localization, different types of navigation, and augmented reality to show informative 3D objects and the surrounding points of interests. Three core processes build this module:

- Getting the user location: In order to get the initial location of the user, the mobile user must find an image marker with the camera. Each marker has a permanent indoor location that is stored in the database. When a marker is detected (see Fig. 3 for an illustrative explanation of the feature or fiducial points of a certain marker), its unique identifier is used to search the database for the location of that marker. Since the user is close to the marker, this location can be assumed to be the user location.
 Updating the user location: Two different methods to update the user location have been implemented. The detection of a new marker will obviously trigger the location update. However, if the user moves away from that location, his/her location is updated with the device sensors that register motion (accelerometer) and direction (compass). This technique is called inertial navigation: the method detects the user footsteps based on the accelerometer readings (see Fig. 4), and updates the user location 2 ft towards the direction read from the compass.

3) Augmented Reality: two types of augmented reality were implemented:

- The 3D Virtual Object AR, which is based on Vuforia SDK [8] and OpenGL ES [9]. Fig. 5 shows a snapshot of the 3D AR objects overlaid on the actual scene captured by the camera. The information of the detected marker (shown in Fig. 3) delivered by Vuforia corresponds to the distance and tilt of the scene, information used to render the 3D object in the “right” place. Moreover, “clickable” virtual buttons, provided also by the Vuforia SDK, were implemented as they allow for visual interaction with the detected image, and used for information retrieval. The 3D object data can be downloadable from the server, this makes the system portable and easily reconfigurable, as the 3D data is obtained from the server on the fly.

- Points of interest AR, shown in Fig. 6. This display method relies on the device sensors, the compass and the gyroscope, to provide a 360-degree view with the points of interest (PoI) in the vicinity of the current mobile user location. As proof of concept and for the sake of simplicity, the PoIs are displayed with a simple clickable button.

4) Database: The database is automatically created and managed on the application using the data supplied by the Internet server. Allowing the iOS to create on the fly the database on the device solves compatibility problems between different iOS releases, and satisfies the application scalability by allowing the addition of new buildings and new databases without needing to modify the code.
5) Search and Finding Directions:

- Locations, points of interest, and users can be searched by typing in the search field (see Fig. 2). Search requests can also be performed with a voice recognition engine, detailed in the next point below.
- Voice-commanded Search. The speech recognition module is optimized to recognize a set of voice commands issued by the user, the recognition is done using the API of iSpeech SDK [10]. It offers the ability to provide the speech recognition with a list of words, allowing the recognition of text that is not available in the dictionary nor the English language, such as Arabic names. It also achieves very good accuracy at recognizing different accents. An example command can be "Where is Dr. Luis Weruaga?".

- Providing Directions: The localization system can be used effectively to obtain the user location and search for other locations. Showing the path from the user location to the destination is the link to providing indoor navigation. In order for the system to search for the path, intersections within the building are added to the location database with links that identify which locations can be directly connected. This data can be used to search for the shortest distance path between any two locations and A* algorithm is the best and most efficient algorithm to use for this type of search. The algorithm guarantees finding the shortest path in the fastest possible way without the need of visiting or testing all locations.

6) Client-Server communication: a lightweight communication protocol was implemented over TCP. The messages are coded in C++, which allows interfacing the same objects in the application and the server code. In order to simplify the designing, coding and sending messages, each message is derived by inheritance from the Message parent class that defines the common header fields. The object of the child class is sent over the network, on the receiving side, the header is used to identify the message type and load the object to its specific class.

B. Server Application

The server application is software running on a PC or workstation. It was implemented using Boost cross-platform C++ libraries, which makes the software to be compiled and run on different operating systems either Windows or various Linux distributions. The server relies on a main database stored in a MySQL server. Fig. 7 shows the server architecture.

The application server provides control information to update the client database, process multiple client requests, and acts as a medium between clients to share their information. Therefore, to make the system intelligent, we can decide the type of information that should be sent to a specific user or group. For example, in a University, students need to know their classrooms, supervisors and public locations available such as the canteen, while the staff may need to know additional information. By making the decision and data customization on the server, which may be hosted on the Internet or locally on the Intranet, enables an intelligent behaviour of the campus.

The clients database can be looked at as cache of the main database, the cached database update mechanism uses timestamps. When the client connects to the server it sends the database timestamp, and as a response, the server checks for new update or entries in the main database and sends only the new data, which saves bandwidth. Furthermore, the server allows pushing information, such that new updates like friends locations can be automatically pushed to the user. However, files like target database, AR objects and the indoor map images are transferred via ftp or http.

IV. EVALUATION

The system was evaluated in the University campus of the authors. The inertial navigation system is reasonably accurate. Fig. 8 shows the estimated path between two points distant more than 300 ft, such that the resulting localization error is less than 2% while holding the telephone in front of the user. A current weakness though is that the precision of the inertial tracking gets influenced by the way the user is holding the phone. We are currently addressing this problem with machine learning techniques such as principal component analysis (PCA). On the other hand, the precision of the trackable detection depends on the features of each trackable image and their uniqueness. Although Vuforia’s image detection algorithm is quite robust against occlusions, we are exploring the effect of removing from the image...
trackable common features present in different trackables, such as the title of the faculty staff member.

Finally, Table I shows a comparison of features available in CAViAR to other mobile applications that provide augmented reality and indoor maps. The combination of features that are included in CAViAR, such as image detection localization with inertial navigation tracking, and AR with indoor localization distinguishes it from other applications.

<table>
<thead>
<tr>
<th>Feature</th>
<th>CAViAR</th>
<th>accessair</th>
<th>junaio</th>
<th>micello</th>
</tr>
</thead>
<tbody>
<tr>
<td>Augmented reality</td>
<td>YES</td>
<td>YES</td>
<td>YES</td>
<td>no</td>
</tr>
<tr>
<td>3D AR Objects</td>
<td>YES</td>
<td>no</td>
<td>YES</td>
<td>no</td>
</tr>
<tr>
<td>Points of interest</td>
<td>YES</td>
<td>YES</td>
<td>YES</td>
<td>no</td>
</tr>
<tr>
<td>Search for location</td>
<td>YES</td>
<td>no</td>
<td>YES</td>
<td>no</td>
</tr>
<tr>
<td>Indoor maps</td>
<td>YES</td>
<td>no</td>
<td>no</td>
<td>YES</td>
</tr>
<tr>
<td>Outdoor maps</td>
<td>YES</td>
<td>yes</td>
<td>no</td>
<td>no</td>
</tr>
<tr>
<td>Indoor localization</td>
<td>YES</td>
<td>no</td>
<td>no</td>
<td>no</td>
</tr>
<tr>
<td>Information sharing</td>
<td>YES</td>
<td>no</td>
<td>no</td>
<td>no</td>
</tr>
<tr>
<td>User Profile</td>
<td>YES</td>
<td>no</td>
<td>YES</td>
<td>no</td>
</tr>
<tr>
<td>Friends</td>
<td>YES</td>
<td>no</td>
<td>no</td>
<td>no</td>
</tr>
<tr>
<td>Locate friends</td>
<td>YES</td>
<td>no</td>
<td>no</td>
<td>no</td>
</tr>
<tr>
<td>Voice command</td>
<td>YES</td>
<td>no</td>
<td>no</td>
<td>no</td>
</tr>
<tr>
<td>Virtual navigation</td>
<td>YES</td>
<td>no</td>
<td>no</td>
<td>no</td>
</tr>
<tr>
<td>Indoor tracking</td>
<td>YES</td>
<td>no</td>
<td>no</td>
<td>no</td>
</tr>
<tr>
<td>Shortest path</td>
<td>YES</td>
<td>no</td>
<td>no</td>
<td>no</td>
</tr>
</tbody>
</table>

Table I

Comparison among CAViAR and Other Mobile Applications.

V. Conclusion and Future Work

This paper presents a new system that can be used efficiently in a University campus to make members aware of their locations, and makes it easier to find places. The use of image detection within the campus enables the system to virtually run on any campus, provided the database for recognizing markers, obtaining the location and displaying the information in AR exists. In addition, inertial navigation provides a suitable approach to track the user location indoors. Furthermore, the system allows students to locate their classrooms, teachers and friends. The user profile should contain data about the student, such as timetable, which can be used to notify students of their next classroom.

In the future, we plan to address the following issues for enhancing the system and improving user experience:

- support for multi locations/multiple floors,
- more accurate tracking, including the combination of wireless localization such as WiFi,
- location detection for multiple identical markers,
- enriching the user profile with more information,
- using profile data to provide user-specific customizable information and services,
- advanced representation of points of interests using graphics, and
- supporting more voice commands.

REFERENCES