

AUGMENTED REALITY-ASSISTED NAVIGATION IN A UNIVERSITY CAMPUS

Vassiliki Kokorogianni¹, Evi Papaioannou^{2*}, Christos Kaklamanis³

¹ University of Patras, GREECE, kokorogian@ceid.upatras.gr

² University of Patras and CTI "Diophantus", GREECE, papaioan@ceid.upatras.gr

³ University of Patras and CTI "Diophantus", GREECE, kakl@ceid.upatras.gr

*Corresponding Author

Abstract

Augmented reality, AR in short, refers to the integration of digital information, like text, multimedia content, GPS data, etc, with the real-world user environment in real time. In this way, as the term implies, the existing, perceived environment is augmented by computer-generated sensory input overlaid in the form of new, complementary information on top of it. Augmented reality differs from virtual reality in the sense that it does not induce a new artificial environment but it enhances existing reality with additional elements of information – real or virtual - of relative semantic content.

Augmented reality was initially used in the context of military, medical and industrial applications, but soon its use expanded into several additional areas including entertainment and games, education, culture, architecture and construction, commerce, advertising. Sophisticated augmented reality applications utilize advanced technologies like computer vision and object/gesture recognition so that users can interact or even digitally alter their real environment. However, simpler augmented reality applications for smartphones exploit GPS and compass functionalities for determining user location and orientation and by means of an augmented reality "marker" they project digital information on the display of the user device.

In this work, we present MyARCampus, an augmented reality application for mobile handheld devices, like mobile phones or tablets. MyARCampus uses image recognition techniques for context awareness and aims to facilitate student life within a university campus. In particular, our application provides detailed information about course timetables for classrooms as well as detailed information about bus timetables for bus-stops located within the Patras University Campus and can be easily extended to cover additional points of interest. Such information is usually provided in the form of printed, static announcements. However, updates and changes are rarely included timely in these printouts. Using MyARCampus, students can obtain anytime useful information about classroom and bus timetables which can highly facilitate course attendance and transportation within the campus and provide invaluable support for efficient schedule organization and management. Furthermore, such an application can be extremely useful, especially in the cases of users who do not speak the local language or do not even understand the signs that are distributed in the deployment location.

Keywords: Augmented reality, application, course and bus timetables, campus

1 INTRODUCTION

Augmented Reality (AR) is a variation of virtual reality (VR). Instead of completely immersing a user inside a synthetic (i.e., virtual) environment with no access to the real world, AR allows the user to be part of the real world which has been supplemented with superimposed or composited virtual objects. In this way, the user receives a perception of the environment where virtual and real objects coexist in the same space.

More precisely, Augmented Reality is a technology fulfilling three key requirements: (a) it combines real and virtual content, (b) it is interactive in real time, and (c) it is registered in 3D. This implies that an AR system must have a display for combining real and virtual images, a computing system for generating interactive graphics which respond to user input in real time and a tracking system for determining the position of the user's viewpoint and enabling the virtual image to appear in the real world (Azuma, 1997, Billinghurst, Clark, Lee, 2014).

Augmented reality enhances the way users perceive or interact with the real world, since virtual objects display information that users cannot directly detect with their own senses. The information conveyed by the virtual objects helps in the performance of real-world tasks (Azuma, 1997). AR can be considered as a particular example of Intelligence Amplification (IA), which refers to the use of a computer as a tool to make a task easier for a human to perform (Brooks, 1996).

Although Augmented Reality has recently become popular, the technology itself is not new. The first computer generated AR experience can be traced back to the 60s due to Ivan Sutherland with "The Sword of Damocles" (Sutherland, 1964, 1968) who received the 2012 Kyoto Prize in Advanced Technology for "pioneering achievements in the development of computer graphics and interactive interfaces"¹. Since then, research and development in AR proceeded until 1980 with early experimentation which provided the AR framework, i.e., the concept together with required technologies. Till the middle 90s, special focus was placed on enabling technologies, like tracking, displays and input devices. Research till 2007 mainly addressed the design and implementation of AR tools and applications exploring interaction methods and usability frameworks. Recently, AR has become a widespread commercial technology evident in a wide range of application areas, like medicine, marketing, gaming, mobile communications (Billinghurst, Clark, Lee, 2014). Initially used in the context of military, medical and industrial applications, AR soon expanded into several additional areas including entertainment and games, education, culture, architecture and construction, commerce, advertising. For instance, recently AR has been used in the context of modern pervasive/ubiquitous computing for urban applications (Chatzigiannakis, Mylonas, Vitaletti, 2011) and outdoors games (Kasapakis, Gavalas, 2017). Sophisticated augmented reality applications utilize advanced technologies like computer vision and object/gesture recognition so that users can interact or even digitally alter their real environment. However, simpler augmented reality applications for smartphones exploit GPS and compass functionalities for determining user location and orientation and by means of an augmented reality "marker" they project digital information on the display of the user device.

In this work, we present MyARCampus, an augmented reality application for smart android devices which uses image recognition techniques for context awareness and aims to facilitate student life within a university campus. In particular, our application provides detailed information about course timetables for classrooms as well as detailed information about bus timetables for bus-stops located within the Patras University Campus and can be easily extended to cover additional points of interest (POI). Such information is usually provided in the form of printed, static announcements. However, updates and changes are rarely included in these printouts. Using MyARCampus, students can obtain anytime online fresh information about classroom and bus timetables which can highly facilitate course attendance and transportation within the campus and provide invaluable support for efficient schedule organization and management.

The rest of the paper is structured as follows: in Section 2, we overview key AR technologies for tracking, display and input. In Section 3, we present design, implementation and usage details for our application, MyARCampus. We conclude and present future plans in Section 4.

2 KEY AUGMENTED REALITY TECHNOLOGIES: TRACKING, DISPLAY, INPUT

Below, we discuss basic enabling technologies for tracking and display, as well as interaction techniques. A detailed recent survey can be found in (Billinghurst, Clark, Lee, 2014).

2.1 Tracking approaches

AR tracking relates to the ability of a system to anchor virtual content in the real world in such a way that it

¹ <http://www.kyotoprize.org/en/laureates/ivan-edward-sutherland/>

eventually appears to be a part of the physical environment. In particular, for superimposing virtual content in the real world, the position and orientation of the viewer must be determined with respect to some sort of "anchor" in the real world.

Depending on the application and involved technologies, the real world anchor may range from a physical object, like a printed image marker, to a particular location determined using some sort of assisting technology, like GPS.

Some of the most popular tracking techniques used for registering virtual content in the real world for the purposes of Augmented Reality include Magnetic Tracking, Vision Based Tracking, Inertial Tracking, GPS Tracking, Hybrid Tracking.

Magnetic trackers are devices that use properties of magnetic fields in order to calculate the pose of a receiver with respect to a transmitter, which is used as the real world anchor.

Vision based tracking exploits a camera and data captured from optical sensors. Vision based tracking has become increasingly popular recently due to very low hardware requirements, improved computational power of user devices and the emergence of ubiquitous computing realized by mobile devices like smart phones and tablets having a camera and screen, equipment which makes them ideal augmented reality platforms.

Inertial tracking uses Inertial Measurement Unit (IMU) sensors such as accelerometers, gyroscopes and magnetometers in order to determine the relative orientation and velocity of tracked objects. Inertial tracking allows the measurement of three rotational degrees of freedom (orientation) relative to gravity. Thus, a change in the position of a tracker can be determined using the inertial velocity and time period between tracker updates.

GPS technology allows for positional tracking in almost all outdoor environments. The current average accuracy of satellite based GPS is less than 3 meters. However, following advancements in satellite technology this accuracy is expected to improve.

Hybrid tracking systems utilize data from multiple sensors for obtaining additional degrees of freedom, improving the accuracy of the individual sensors and overcoming weaknesses of other single tracking methods.

The most widely used tracking approaches currently use computer vision techniques and can provide both marker-based and marker-less tracking. Furthermore, mobile devices, like smartphones and tablets, offer new opportunities for hybrid tracking. Many mobile devices are equipped with cameras, accelerometers and gyroscopes, GPS and wireless networking capabilities which can be combined for obtaining accurate pose estimation for both indoor and outdoor AR tracking (Billinghurst, Clark, Lee, 2014).

2.2 Display approaches

Display technologies enable the combination of real and virtual images so that both are seen simultaneously. Combined with AR tracking, AR displays create the illusion that virtual content is really part of the real world. There is a variety of different types of displays that can be used to combine real and virtual images. Combining images of the real and virtual scenes for AR visualization requires the performance of tasks like camera calibration, registration, tracking and composition.

While AR displays use different approaches to combine the real and virtual views, they can be categorized based on the placement of the display is placed between the user and the real world scene. Head-attached displays present virtual images right in front of the user eyes. Handheld or body-attached displays are considered to be mobile and personal, yet sharable. They are also more socially acceptable compared to head-attached displays. Spatial displays feature limited mobility and are usually installed at a fixed location. As spatial displays tend to provide a larger image in many cases, they are more applicable as public displays sharable by multiple users. While most of the research in AR focuses on visual augmentation of the real world, there is also work on AR displays for the other senses, such as audio or haptic. These types of non-visual displays are mostly used in Virtual Reality (VR) where the main focus is to replace all human sensory input with computer generated virtual signals.

Although most early AR systems used head-mounted displays, the most common display type currently used is a handheld device such as a mobile phone or tablet. However this may change with the emergence of lower cost head-wearable computers and displays in combination with ongoing research in different display systems.

2.3 Input approaches

AR systems can incorporate various types of input methods, ranging from traditional 2D user interfaces (UI), e.g., keyboard, mouse and touch screen, to 3D and multimodal interfaces, e.g., handheld sticks, speech, gesture, etc.

Different input methods have been used for different types of AR applications depending on the required user interaction tasks. So, the interface type used has evolved over time. Since the 1960's, several AR interface types have been developed, including:

- Information Browsers, i.e., interfaces for projecting AR information on the real world
- 3D User Interfaces which use 3D interaction techniques to manipulate content in space and controllers in order to interact with virtual objects
- Touchable User Interfaces, which use real objects for interaction with AR virtual content
- Natural User Interfaces, which use body input like hand gestures
- Multimodal Interfaces, which use a mixture of speech and gesture input

Simple information browsing supports very little interaction with the virtual content, while complex interfaces can recognize and respond to natural gestures and speech. However, even though several input methods are now available, they are not all appropriate for all types of applications.

2.4 Development tools

There are various tools for implementing AR systems that can be arranged in a hierarchy of decreasing programming skills required to use them; from low level software libraries that require developers to have significant programming skills, to simple authoring tools that require no programming ability.

Various types of AR development and authoring tools are available. Low level software libraries and frameworks provide high flexibility but require advanced programming skills. On the other hand, standalone authoring tools enable end users to create their own – rather simple - AR content easily with minimal computer skills. ARToolkit, MXRToolKit, osgART, Studierstube are examples of low-level software libraries/frameworks which require advanced programming and coding skills. FLARManager, Processing, OpenFrameworks are examples of tools for rapid prototyping which require elementary programming knowledge but advanced design/prototyping skills. AR-Media plug-ins, DART, Metaio Unity, Vuforia can be used with other development tools and require experience with the developer tool the plug-in works with. BuildAR, Metaio Creator, Layar Creator, Wikitude Studio are examples of standalone AR authoring tools which require no former programming knowledge.

AR development tools also have different target application platforms. Some tools are suitable for developing desktop AR applications while others are suitable for mobile AR application. Some tools support tracking based on computer vision for developing AR applications mainly for indoor use while others exploit sensors on a mobile device for developing outdoor AR application.

While commercial authoring tools and AR plug-ins for game engines are widely adopted, the main interface of these authoring tools is still based on 2D graphical user interfaces which impose particular limitations regarding the manipulation of 3D content. As the technology advances, it is expected that the authoring and development tools will more actively adopt 3D and AR interfaces.

At this point, it is worth mentioning the technologically inspiring and visionary approach that was presented very recently by M. Zuckerberg² at the Facebook developer conference-F8³. The camera is planned to be the first augmented reality platform. Recent AR trends focus on glasses or eventually contact lenses that look and feel normal enabling the overlay of all kinds of information and digital content on the real world. But in the last couple of years, primitive examples of AR on phones in the camera have appeared. In the near future, developers will have the tools to build all kinds of AR tools into a camera, extending the physical world online. AR is going to help to mix the digital and the physical in new ways and make physical lives better.

3 MYARCAMPUS

MyARCampus is an augmented reality application for smart android devices based on image recognition. It

² https://en.wikipedia.org/wiki/Mark_Zuckerberg

³ <https://www.fb8.com/>

aims to facilitate student life within a university campus. In particular, our application provides detailed information about buildings and course timetables for classrooms therein as well as detailed information about bus timetables for bus-stops located within the Patras University Campus. MyARCampus can be easily extended to cover additional points of interest (POI). Such information is usually provided in the form of printed, static announcements. However, updates and changes are rarely included in these printouts. Using MyARCampus, students can obtain anytime (online) fresh information about classroom and bus timetables which can highly facilitate course attendance, navigation and transportation within the campus and provide invaluable support for efficient schedule organization and management.

3.1 Hardware specifications

We implemented our application using the Unity programming environment. In particular, we used Unity version 5.5.2f1 (personal) running on a 64-bit Windows 10 Home machine with an Intel(R) Core(TM) i5-5200U CPU@2.20GHz and a RAM of 8GB.

In Table 1, devices where our application was exported and tested are presented.

Table 1: Devices where our application was exported and tested

Device	Smartphone	Tablet
Model:	Xiaomi Redmi note 4	Samsung Galaxy Tab E
Android Version: Sdk:	6.0 (marshmallow) 23	4.4.4 (KitKat) 19
CPU:	Deca-core Max 2.11GHz, ARM Cortex-A53	Quad-core, 1300 MHz, ARM Cortex-A53
RAM	3.0 GB	1.5 GB
Battery	4100mAh	5000mAh
Kernel Version:	3.18.22+	3.10.17-1167403
Screen size:	5.5 inch	9.6 inch
Screen Resolution: Screen Density:	800 x 1280 pixels 157 ppi	1080 x 1920 pixels 401 ppi
OpenGL ES	3.0	2.0

In order to execute and exploit MyARCampus, users must have access to an Android device with the following minimum system requirements:

- ARMv7 (Cortex family) Processor (or later)
- The device GPU must support OpenGL ES 2.0 (or higher)
- OS: Android 4.0.3 ('Ice Cream Sandwich') (or later)
- Rear-facing camera

3.2 Software platforms and packages

For the development of MyARCampus we used the Vuforia Software Development Kit (SDK)⁴. Compared to other available Augmented Reality SDKs, Vuforia provides a complete and extremely helpful support system. In particular, the Vuforia development portal contains an integrated forum where users and developers cooperate, discuss and suggest solutions to various technical issues. Furthermore, Vuforia provides a straightforward setup process. Detailed information is provided regarding the setup of the environment, the installation of the appropriate SDK version and building a sample application. Additionally, it is free to use for developers and, as a Unity extension, it consists of a simple drag-and-drop authoring workflow.

A software development environment or integrated development environment (IDE) was also necessary for the development process. We used the Unity⁵ engine (version 5.5.2f1/personal) due to a number of reasons which include its compatibility with the Vuforia library and the Android system, its fast-performing assembly runtime, pain-free deployment process and satisfactory documentation. Another factor favoring the selection of Unity stemmed from the fact that it certainly requires less coding compared to other popular platforms, like Android Studio, which makes it an attractive environment for efficiently developing useful AR applications requiring reasonable time complexity and programming skills.

⁴ <https://www.vuforia.com/>

⁵ <https://unity3d.com/unity/>

As far as technicalities are concerned, since Vuforia is only a library, it is necessary to first install Unity and then import the Vuforia Unity Extension (available the Vuforia official website) into the Unity project. Also, Android development tools (available at the Android SDK site⁶) must be installed. At this point, it is very important that the correct drivers for our Android device are installed and that USB Debugging is enabled on our Android Device.

3.3 Developing with Target Images

Our application is based on the Image Targets sample project from the Vuforia website. We decided to use Image Recognition instead of location-based Augmented Reality mainly due to implementation efficiency supported by sophisticated Vuforia algorithms.

However, GPS (Global Positioning System) reliability of smartphones also played an important role in favor of the use of target images. More precisely, it has been widely observed that location data accuracy comes down to an interplay between several factors, including signal source (GPS signals, Wi-Fi, cell tower triangulation), environment (area density, skyline view, indoor or outdoor location) and personal use (location data access enabled, type of mobile app used, operating system usage). Therefore, instabilities in device performance could raise reliability and accuracy issues which could negatively affect our application. Nevertheless, location and map based services can be easily integrated into MyARCampus, to embed additional features, extend functionality and enrich user experience.

Regarding the practical part of the procedure, developing with Image Targets required us to first capture photographs of various points of interest (POI) within the university campus, such as bus stops, classroom and building entrances etc. Then, we proceeded to the creation of a new database on the Vuforia Target Manager where we uploaded captured images. Finally, we downloaded our database and added it to our project using the "Import Package" and "Custom Package" facilities of the Unity Project.

3.4 Adding AR assets and prefabs to a scene

Henceforth, adapting our project to use augmented reality involves no particular complexity. First, an instance of the AR Camera prefab from the Vuforia/Prefabs folder must be dragged into a scene. The AR Camera is responsible for rendering the camera image in the background and manipulating scene objects to react to tracking data. Then, an instance of the Image Target prefab must be dragged into the scene. This prefab represents a single instance of an Image Target object; at the Inspector of the selected Image Target, the data set it belongs to (database) and its specific instance (photo) must be selected.

Instead of the usual 3D objects, our application uses popup windows which appear whenever a target is detected. We decided to use popup windows instead of – probably more fancy but equally functional – 3D objects in order to produce simple-to-use and light-weighted AR application. Popup windows can be created in Unity by using Hierarchy lists for attaching selected content onto the canvas of Image Target items. The "pop-up" effect can be achieved via the development of suitable C# scripts.

For enabling MyARCampus to detect several different points of interest, that is, multiple image targets, we enhanced Unity scenes with several Image Target prefabs appropriately assigning them desirable popup windows.

3.5 Using MyARCampus

For exploiting MyARCampus, users must install it on an Android device (smartphone or tablet), run it and hold their device horizontally looking around with the camera. Whenever a point of interest (POI) is recognized by the application, a pop-up window appears on top of the camera identifying the POI together with a button which either redirects to a certain website or to a new window or file containing information about the POI they are currently looking at.

Currently, there are three types of POI included in MyARCampus: 1) buildings and complexes, 2) classrooms, auditoriums, libraries, 3) bus stops. When a building is recognized, users can access the official website of the corresponding department (see for example Fig. 1). When a classroom or auditorium is identified, users can view its weekly schedule (see for example Fig. 2 and Fig. 3). When libraries, computer centers or other facilities are identified, users can view useful information, like timetables, staff, contact info, etc (see for example Fig. 4). When a bus stop is identified, users can view the available bus lines, their itinerary and corresponding timetables (see for example Fig. 5 and Fig. 6).

⁶ <https://developer.android.com/index.html>



Fig. 1: When a building is recognized, users can visit the official website of the corresponding department.

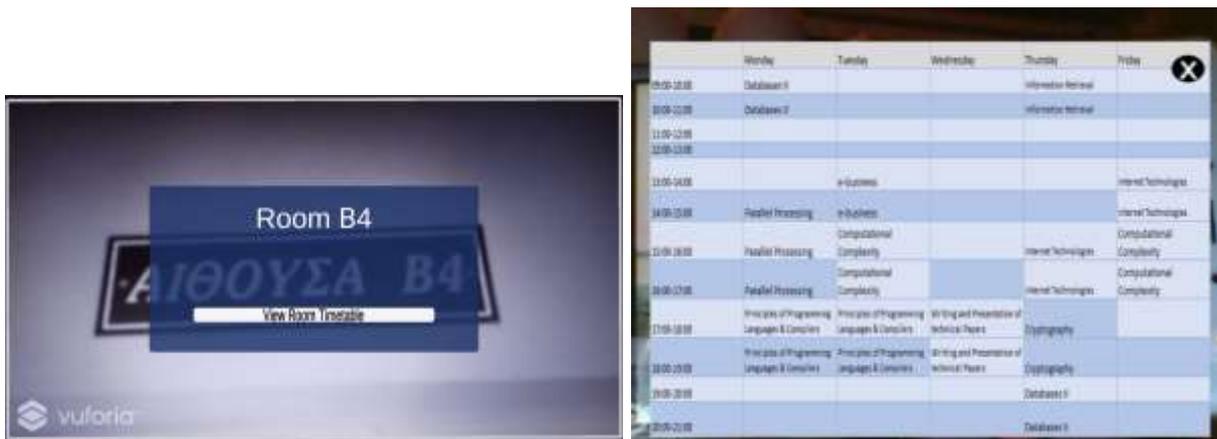


Fig. 2: When a classroom is identified, users can view its weekly schedule.



Fig. 3: When a. auditorium is identified, users can view its weekly schedule.

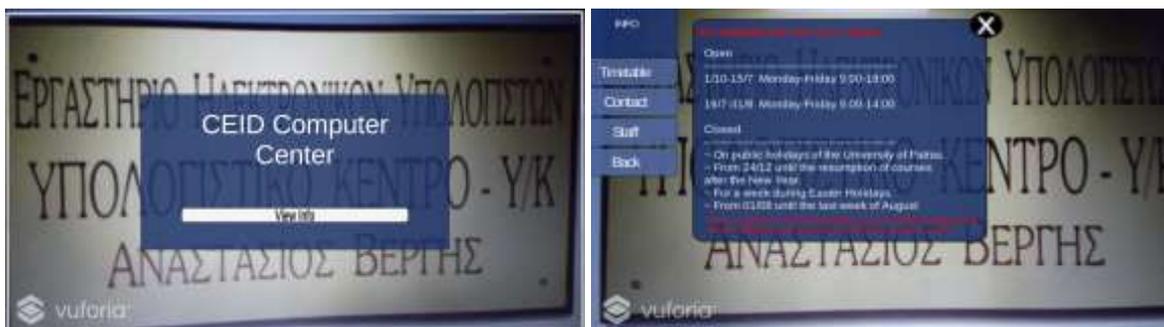


Fig. 4: When facilities are identified, users can view useful information for e.g. timetables and contact.



Fig. 5: A bus stop (a type 3 point of interest (POI)) has been identified.



Fig. 6: When a bus stop is identified, users can view available lines, itineraries and timetables.

MyARCampus  can be downloaded and installed either directly from <https://drive.google.com/drive/folders/0B3hxDillGpvgY205cUpuNWhodmM> or, alternatively, from mobango (<http://www.mobango.com/>) or getjar (<https://www.getjar.com/>) (see Fig. 7).

The application is free and ready to use. Registering as a user is not required. For demonstration and testing purposes, captured pictures from our POI are also available⁷.

⁷ <https://drive.google.com/drive/folders/0B3hxDillGpvgLVZwUW9fbm13ZEK>

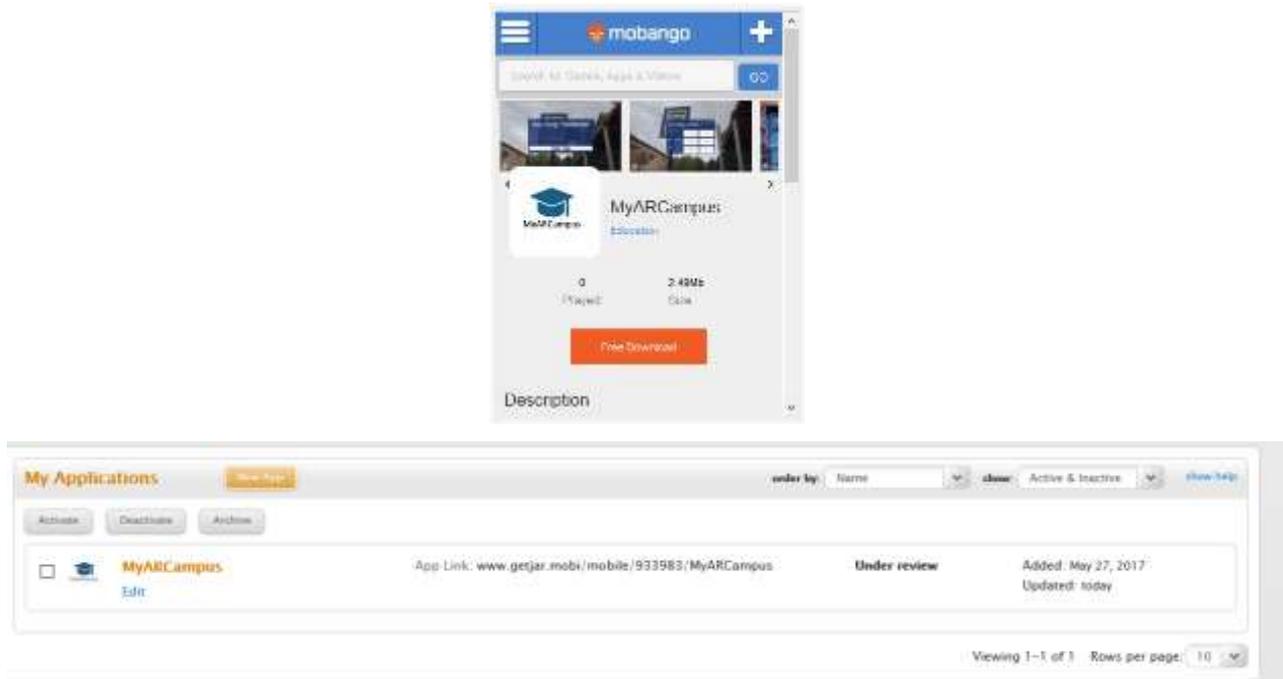


Fig. 7: MyARCampus at mobango and getjar app repository.

So far, MyARCampus has been used and evaluated by a closed group of students and personnel of our department. Current feedback shows that it is indeed considered to be a very useful, easy-to-use application which supports administration and facilitates student life within the campus, making schedule organization and management easier and more efficient.

4 CONCLUSION AND FUTURE PLANS

We designed and implemented MyARCampus, a augmented reality application based in image recognition which via a handheld device such as a mobile phone or tablet provides detailed information about buildings, course timetables for classrooms as well as detailed information about bus timetables for bus-stops in a university campus. Such information is usually provided in the form of printed, static announcements or is accessible at specific websites. However, updates and changes are rarely included in time. Using MyARCampus, students can easily obtain useful information about classroom and bus timetables which can highly facilitate course attendance and transportation within the campus and provide invaluable support for efficient schedule organization and management. Furthermore, MyARCampus as well as relevant AR applications can be extremely useful, especially in the cases of users who do not speak the local language or do not even understand the signs that are distributed in the deployment location

As far as our next steps are concerned, it is within our plans to extend the current version of MyARCampus so that it covers a wider area of our campus, offering additional forms of information, highly visualized where possible. We also plan to upgrade MyARCampus graphical interface at the lowest possible cost on performance and user-friendliness. Our future work will further exploit pop-up windows as a way of simplifying the usability of the solution. In particular, we plan to work on the design and implementation of an extension of MyARCampus for visually-impaired persons by using, for example, device vibration to alert the user that a POI has been identified, along with text-to-speech technologies to substitute “pop-ups”.

Finally, we plan to promote the use of MyARCampus in order to obtain valuable feedback from the experiences of additional users and, most importantly, for contributing to making a University campus an attractive, easy-to-explore and inspiring place to live, create and evolve.

5 ACKNOWLEDGEMENT

We would like to thank Prof. Charalampos Konstantopoulos, Prof. Damianos Gavalas and Natasa Ioannidi (MSc) for their valuable help and support.

REFERENCE LIST

- Azuma, R.T. (1997). A Survey of Augmented Reality. *Presence*, vol. 6, no 4, pp. 355-385, MIT Press.
- Azuma, R., Baillot, Y., Behringer, R., Feiner, S., Julier, S., and MacIntyre, B. (2001). Recent advances in augmented reality. *Computer Graphics and Applications*, IEEE, vol. 21, no 6, pp. 34–47.
- Billinghurst, M., Clark, A., and Lee, G. (2014). A Survey of Augmented Reality. *Foundations and Trends in Human-Computer Interaction*, vol. 8, no. 2-3, pp. 73–272.
- Brooks, F. P., Jr. (1996). The computer scientist as toolsmith II. *Communications of the ACM*, vol. 39, no 3, pp. 61-68.
- Carmigniani, J., Furht, B., Anisetti, M., Ceravolo, P., Damiani, E., and Ivkovic, M. (2011). Augmented reality technologies, systems and applications. *Multimedia Tools and Applications*, vol. 51, no 1, pp. 341-377.
- Chatzigiannakis, I., Mylonas, G., Vitaletti, A. (2011). Urban pervasive applications: Challenges, scenarios and case studies. *Computer Science Review*, vol 5, no 1, pp. 103-118.
- Kasapakis, V., Gavalas, D. (2017). Occlusion handling in outdoors augmented reality games. *Multimedia Tools and Applications*, vol. 76, no 7, pp. 9829-9854.
- Sutherland, I. E. (1964). Sketch pad a man-machine graphical communication system. In *Proceedings of the SHARE Design Automation Workshop*, ACM, pp. 6–329.
- Sutherland, I. E. (1968). A head-mounted three dimensional display. In *Proceedings of the 1968 Fall joint Computer Conference (AFIPS)*, part I, ACM, pp. 757–764.
- van Krevelen, D.W.F. and Poelman, R. (2010). A Survey of Augmented Reality Technologies, Applications and Limitations. *International Journal of Virtual Reality*, vol 9, no 2, pp. 1-20.
- Zhou, F., Been-Lirn Duh, H., and Billinghurst, M. (2008). Trends in augmented reality tracking, interaction and display: A review of ten years of ismar. In *Proceedings of the 7th IEEE/ACM International Symposium on Mixed and Augmented Reality*, IEEE Computer Society, pp. 193–202.