International Conference on Innovations in Engineering and Industrial Applications (ICIEIA ’18)

Smart home interaction using Augmented Reality with Internet of Things

LV Abarajithan[a]

[a] *Department of CSE, K.L.N College of Engineering, Pottapalayam, Madurai*

Abstract

Augmented Reality (AR) technology constitute a suitable solution for IoT interaction. By implementing Augmented Reality technology to smart phones, the interaction with IoT is made better. Augmented Reality enhanced the natural environments or situations and offered perceptually enriched end-user experiences. It made the interaction with physical reality more responsive. By not just using a Graphical User Interface (GUI) based interaction such as a button or slider, a real-world touch or gesture on virtually placed objects was implemented to control IoT devices. Smart phone was pointed out to a hardcoded 2D or 3D object provided a virtually placed controller (buttons etc.,) on that object, which enabled to control the connected IoT devices with a virtual touch or a random gesture on that controller. When the actions were triggered, the state of the IoT enabled devices were sent or stored in a cloud database. The IoT device was able to read the state from the cloud storage database and change its state accordingly. By this type user interaction, AR plays an attractive visual interface for IoT devices and IoT acts as a basis for scaling AR technology.

Keywords: Internet of things (IoT); Augmented Reality (AR); Smart home;

1.Introduction

The Internet of things (IoT) is the network of physical devices, vehicles, home appliances and other items embedded with electronics, software, sensors, actuators, and connectivity which enables these objects to connect and exchange data. Each thing is uniquely identifiable through its embedded computing system but is able to inter-operate within the existing Internet infrastructure. Augmented reality (AR) is a direct or indirect live view of a physical, real-world environment whose elements are "augmented" by computer-generated perceptual information, ideally across multiple sensory modalities, including visual, auditory, haptic, somatosensory, and olfactory. The overlaid sensory information can be constructive (i.e. additive to the natural environment) or destructive (i.e. masking of the natural environment) and is spatially registered with the physical world such that it is perceived as an immersive aspect of the real environment.

In this way, Augmented reality alters one’s current perception of a real world environment, whereas virtual reality replaces the real world environment with a simulated one. Augmented Reality is related to two largely synonymous terms: mixed reality and computer-mediated reality.

IoT devices are a part of the larger concept of home automation, also known as domotics. Large smart home systems utilize a main hub or controller to provide users with a central control for all of their devices. These devices can include lighting, heating and air conditioning, media and security systems (including access control systems, namely August, Ausweis.io, Kwikset, Schlage). Ease of usability is the most immediate

[a] * LV Abarajithan. Tel.: +91 9543372557.
E-mail address: lvabarajithan@gmail.com.

benefit to connecting these functionalities. Long term benefits can include the ability to create a more environmentally friendly home by automating some functions such as ensuring lights and electronics are turned off. One of the major obstacles to obtaining smart home technology is the high initial cost.

Augmented reality applications, running on handheld devices utilized as virtual reality headsets, can also digitalize human presence in space and provide a computer generated model of them, in a virtual space where they can interact and perform various actions. Such capabilities are demonstrated by "Project Anywhere", developed by a postgraduate student at ETH Zurich, which was dubbed as an "out-of-body experience". AR on smart phones will lead the way, due to three things: hardware that we already own and use; an almost infinite number of potential applications; and jaw-dropping, genuinely practical use-cases that will transform the way we use our smart phone.

2.Related Works

In early days, AR systems had been implemented as single applications (using APIs and toolkits) with only a limited number of objects targeted for augmentation. Data necessary for recognition and associated contents were all pre-included within the application. AR system architecture has now evolved into a two tier structure, divided into (1) a server taking the role of recognizing and tracking targeted real objects against those stored in the database and (2) a client application (or an AR browser using a separate content server) associating user, service, or context specific contents with the recognized and tracked target.

The database part of the server-centric AR system stores mainly two types of information; one is the object identity and a set of essential features used by a recognition or tracking algorithm (by the same server), and the other is the associated augmentation contents with the contexts for them to be invoked. A recent trend in the field of AR is to do away with artificial markers (or any attachments to the target object) that are less appealing (e.g., large thick black boundaries with geometric patterns), and instead be able to recognize, track, and augment everyday objects in their original forms.

When a user enters such an environment, the presence of nearby IoT capable objects is detected through standard wireless service discovery protocols. After additional message exchanges, the AR capable objects can be further identified, and even filtered out based on their relative distances or directions from the user

Sixth Sense, which could be considered as a mobile version of a projected AR UI. It consists of a camera and a small projector mounted on a hat or coupled in a pendant. The camera tracks the user’s hand gestures and the projector visually augments virtual content on the physical objects that the user is interacting with. But again there are severe limitations, as it is built by several parts that are bulky to use and it requires a planar surface in front of the user to display content.

A recent trend in the field of AR is to do away with artificial markers (or any attachments to the target object) that are less appealing (e.g., large thick black boundaries with geometric patterns), and instead be able to recognize, track, and augment everyday objects in their original forms. Using natural features is the most prominent method of recognition and tracking of marker-less objects in AR today. However, the method relies on a minimum number of features, typically several hundreds, that remain invariant under different usage situations (e.g., lighting or pose) and may not perform well for texture-less objects.

Using modern consumer-oriented AR glasses to prototype AR UIs overcomes many of these limitations, and therefore, the study described in this paper utilized Microsoft HoloLens for the purpose of prototyping. One notable example of a large scale AR service that is also based on a server-client architecture, is the location or GPS based guidance for geographical locations or places of interest (e.g., the nearest restaurant, drug store, tourist sites). In this case, despite its large scale coverage in terms of the physical area and number of users serviced, the amount of associated data utilized in this service is much smaller (e.g., GPS coordinates) with a significant amount of the data (e.g., the map of the area) already downloaded onto the client device ahead of time. In addition, there is no object or place specific tracking involved. Therefore, in ARIoT, object-specific information for recognition and tracking, including the preferred tracking method, Whereas AR has recently been touted as one of the ideal interfaces for IoT, no notable and mature work has been conducted in this area. On a similar note, AR has been considered as an interface to physical and plain objects in different forms. A few comparative studies have been conducted to validate the effectiveness of AR as an interface for manuals and instruction guidance. Ullah et al presented a remote-touch system for smart home control in which the user would touch the augmented object seen on a display. There have been other similar approaches, all implemented as unique applications that have not addressed scalability (e.g., handling hundreds or thousands of objects).

3.AR Interface Implementation

Augmented Reality adds on to the basic IoT environment with objects set up to contain basic and generic information such as their object IDs, button IDs, status, and control interfaces by the object owners. In addition, objects are supplied with “features” and “contents” for the purpose of AR services.

When a user points his/her smartphone onto the target object, a virtual button appears next to it. This button is placed on the real-world spatial area virtually, so that the user sees it like it is physically present. This Virtual button acts as the controller of that target IoT device. The user can interact with the button, by a touch or gesture, to control the state of the IoT device. When the button detects touch or gesture, actions are triggered and based on the current state of the IoT device, the status is updated onto the real-time cloud storage. Figure 1 shows the feature points on a target image.

Fig. 1. An Image target as masked image and with feature points

Equations Since these interaction happens in real-time, a real-time cloud database is used to save the state of the device and notify it. By this way, when ever status of the device is updated on the cloud, the device automatically gets notified that a change has been detected, in realtime. The object detection is based on the feature points present on the target. The higher the feature points, the better the sensitivity of the virtual button. The virtual button is placed on the target, where there is enough amount of feature points. Thus, when ever the smart phone camera detects any disturbance in the feature points, it means that a virtual press or release action is detected. Figure 2 shows the virtual buttons on an image target.

It is to be noted that the information exchange between the AR client and the IoT objects can occur directly. This process is expected to be fast and robust as there are only a few candidate target objects to look for. Again, different IoT objects may contain different types of feature information (and the corresponding recognition algorithm type) depending on their characteristics. This is called the “guided” recognition and tracking applied to natural features such as textures and corners of rich feature objects.

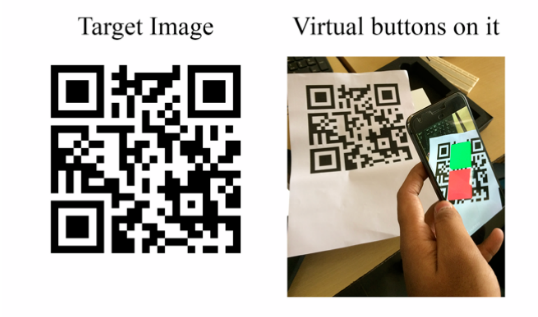

Fig. 2. A Target Image (Left), Virtual buttons on target image after recognized by smart phone camera (Right)

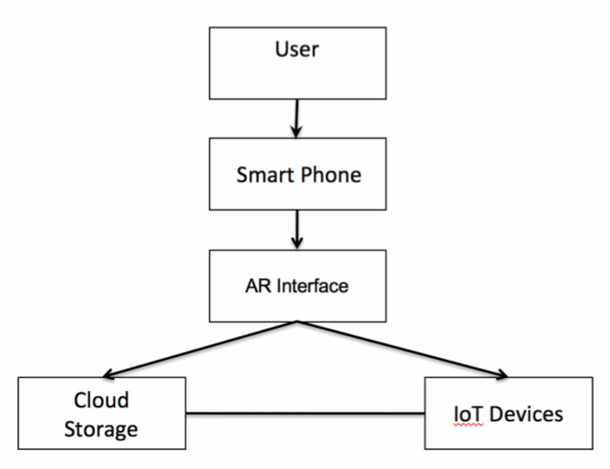

Fig. 3. Block Diagram

A user interacts with the IoT devices, which is connected to cloud storage, through Augmented Reality interface with a smart phone camera. The Smart phone is connected to the cloud storage database which sends the state of the IoT device to the cloud. The IoT device reads the state from the cloud storage database and change its state accordingly.

4.Framework

4.1.Implementing AR Interface

Image targets database consists of hardcoded planar images or 3D objects such as a box. These targets database is imported into Unity3D tool. Each target in the database consists of recognizable target points features. Virtual buttons provide a useful mechanism for making image based targets interactive. The size and placement of Virtual buttons must be considered when designing an experience that uses them. There are several factors that will affect the responsiveness and usability of Virtual buttons.

- The length and width of the button
- The area of the target that it covers

• The placement of the button in relation to the both the border of the image, and other buttons on the target

Virtual Buttons detect when underlying features of the target image are obscured from the camera view. We need to place the button over an area of the image that is rich in features in order for it to reliably fire its event.

After Importing the database, an image target needs to be drawn on the Unity Scene. Two or more virtual buttons are placed on top of the image target where there are higher features points. The higher the feature points, the better the sensitivity. The C# script is written for the current scene.

Fig. 4.1. A Target Image (Left), Virtual buttons on target image after recognized by smart phone camera (Right)

4.2.Connecting to Cloud Storage

A new project is created on the Firebase console and the required configuration JSON file and the Software Development Kit (SDK) is imported into the Unity 3D tool. Implementation of the Firebase Realtime cloud storage database involves 3 phases –

1. Integrate the Firebase Realtime Database SDKs- Download the SDK and import it into the Unity 3D tool using Project view. The configuration JSON file is placed appropriately inside the project folder.

2. Create Realtime Database References- Firebase is a JSON database which is referred using path references. The path reference is appropriately chosen (Eg. users/user:1234/phone_number) in C# script. This reference can be used to set data or subscribe to data changes at that particular database reference.

3. Add Event Listener- Firebase data is written to a Firebase Database reference and retrieved by attaching an asynchronous listener to the reference. The listener is triggered once for the initial state of the data and again anytime the data changes. This listener is triggered once when it is attached and again every time the data, including children, changes. This event listener is used by the IoT devices to read for changes. When a Virtual button event is triggered, the data can be written on to this reference. The change is notified to the IoT device immediately, in real-time, so that it can change its state appropriately.

```
compile 'com.google.firebase:firebase-database:12.0.0'
```

Fig. 4.2. Android Gradle dependency for Firebase Realtime database

4.3.Setting up IoT devices

An NXP Pico i.MX7D with latest image is flashed and is booted up. the Android Things Console provides tools to install and update the system image on supported hardware devices.

Things apps use the same structure as those designed for phones and tablets for Android. This means that we can modify existing apps to also run on embedded things or create new apps based on what we already know about building apps for Android. Two or more led's are connected to any one of GPIO pins. A display is attached to show the current state of the connected components.

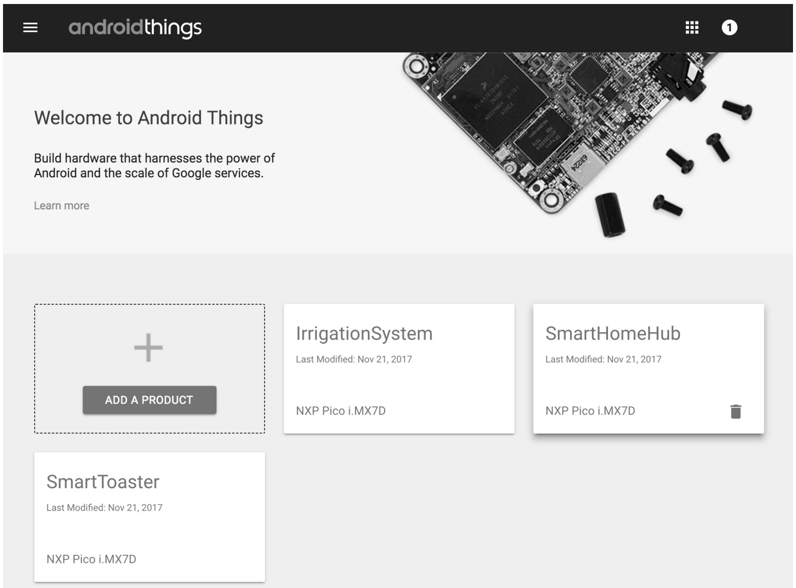

Fig. 4.3. Android Things Console

In Android studio, a new project is created for Android app. From Firebase console, the configuration file and Firebase SDK is imported via gradle plugin in Android Studio. Implementation of the Firebase Realtime cloud storage database involves 3 phases –

1. Integrate the Firebase Realtime Database SDKs- Download the SDK and synced into the Android Studio via gradle plugin. The configuration JSON file is placed appropriately inside the project folder.

2. Create Realtime Database References- Firebase is a JSON database which is referred using path references. The path reference is appropriately created (Eg. users/user:1234/phone_number) in Java. This reference can be used to set data or subscribe to data changes at that particular database reference.

3. Add Event Listener- Firebase data is written to a Firebase Database reference and retrieved by attaching an asynchronous listener to the reference. The listener is triggered once for the initial state of the data and again anytime the data changes. This listener is triggered once when it is attached and again every time the data, including children, changes. When a Virtual button event is triggered, the data is written on to this reference. The change is notified to the listener, in real-time, so that it can change its state appropriately.

5.Results and Discussion

5.1.IOT Kit

The NXP Pico i.MX7D board is setup to connect to the internet via Wi-Fi module. Once it is connected to the internet, the device starts listening to the realtime cloud database. When ever any change is detected in the cloud storage database, the state of the IoT device is changed accordingly.

Fig. 5.1. NXP Pico i.MX7D board

5.2.AR Interface

The Smartphone AR app recognizes the target object and places virtual buttons on it. This buttons can be interacted with a virtual touch or a gesture to trigger a state change action on the IoT device. When any action is triggered, the state is written to the cloud storage database and hence, the IoT device which is listening in realtime will change its state accordingly.

Fig. 5.2. AR Interface on Smartphone

5.3.Interaction

When a virtual touch or gesture is made over the blue button, the Smartphone app (which is connected to the cloud) sent the state information as “ON” to the cloud storage database.

The IoT device, which is listening to the cloud, changes the state of the device to state “ON”.

Fig. 5.3. Device state change on LED

6.Conclusion

Augmented Augmented Reality is a physically context aware interface for Smart Home interaction using Internet of Things. In many cases the GUI (Graphical User Interface) of the mobile is difficult to use for the disabled people. This system uses Augmented Reality (AR) technique for interfacing which is getting increasingly popular for controlling of home appliances. Using this technology, it will greatly help the disabled and elderly people to control the appliances easily. Different virtual switches will appear when the camera is pointed to different appliances, thus allowing the user to control different appliances easily and conveniently. Instead of 2D buttons, 3D switches will appear on the screen which gives a familiar interface to the user. ARIoT is capable of providing a much friendlier environment to designate an object and associate augmentation contents including object control of home appliances, and improving the overall interactivity of the IoT environment. The AR client can be implemented easily using simple algorithms owing to the small number of objects to recognize and track, and thus be self-contained within the mobile device.

References

Dongsik Jo and Gerard Jounghyun Kim “ARIoT: Scalable Augmented Reality Frameworkfor Interacting with Internet of Things Appliances Everywhere” Vol. 62, No. 3, August 2016.

Bejin Oh and Woontack Woo ,‘‘CAMAR: Context-aware Mobile Augmented Reality in Smart Space’’ IWUVR 2009, pp. 48-51, 2009.

Cünter Alce, Maximilian Roszko, Henrik Edlund, Sandra Olsson, ” AR as a User Interface for The Internet of Things—Comparing Three Interaction Models," IEEE/ACM 9-13 Oct. 2017.

D. Díaz-Sánchez, F. Almenarez, A. Marín, D. Proserpio, and P. A. Cabarcos, “Media cloud: An open cloud computing middleware for content management,” IEEE Trans. Consumer Electron., vol. 57, no. 2, pp. 970-978, May 2011.

A. Dame and E. Marchand, “Video mosaicing using a Mutual Information-Based Motion Estimation Process,” in Proc. IEEE Int. Conf. on Image Processing, pp. 1525-1528, 2011.

D. Do and G. J. Kim, “In-situ AR manuals for IoT appliances,” in Proc. IEEE International Conference on Consumer Electronics, pp. 411-412, 2016.

R. Gimenez and M. Pous, “Augmented reality as an enabling factor for the Internet of Things,” W3C Workshop: Augmented Reality on the Web Mobile 2.0, Barcelona, Spain, 2010.

L. Guller, I. Aslan, and L. Krußen, “GuideMe: A mobile augmented reality system to display user manuals for home appliances,” in Proc. Advances in Computer Entertainment, pp. 152-167, 2013.

F. P. Hrooks, “The computer scientist as toolsmith,” Communications of the ACM, vol. 39, no. 3, pp. 61-68, Aug. 1996.

E. Hsiao and M. Hebert, “Occlusion Reasoning for object detection under arbitrary viewpoint,” in proc. IEEE Conf. Computer Vision and Pattern Recognition, pp. 3146-3153, 2012.

E. Hsiao, A. Collet, and M. Hebert, “Making specific features less discriminative to improve point-based 3d object recognition,” in Proc. Computer Vision and Pattern Recognition, pp. 2653-2660, 2010.

Z. Iang, W. Li, P. Hui, and C. Peylo, “CloudRidAR: A cloud-based architecture for mobile augmented reality,” In Proc. Workshop on Mobile Augmented Reality and Robotic Technology-based Systems, pp. 29-34, 2014.

B. R. Juang, C. H. Lin, and C. H. Lee, “Mobile augmented reality based on cloud computing,” in Proc. International Conference on Anti-Counterfeiting, Security and Identification, pp. 1-5, 2012.

S. Kasahara, V. Heun, A. S. Lee, and H. Ishii, “Second surface: Multi- user spatial collaboration system based on augmented reality,” in Proc. SIGGRAPH Asia 2012 Emerging Technologies, pp. 1-4, 2012.

M. Lechner, “ARML 2.0: An overview,” 93rd Technical Committee in Open Geospatial Consortium, 2014.

Y. Ochen, L. Xiang, J. Zhang, and L, Liu, “Research about mobile AR system based on cloud computing,” in Proc. Wireless and Optical Communication Conference, pp. 355-359, 2013.

C. Perey, “Open and interoperable augmented reality and the IEEE standards,” IEEE Consumer Electron. Magazine, vol. 4, no. 4, pp. 133- 135, Oct. 2015.

R. Radkowski, “Augmented reality to supplement work instructions,” Model-Based Enterprise Summit, 2013.

J. Rekimoto, and Y. Ayatsuka, “CyberCode: Designing augmented reality environments with visual tags,” In Proc. Of Designing Augmented Reality Environments, pp. 1-10, 2000.

C. Suh and Y. B. Ko, “Design and implementation of intelligent home control system based on active sensor networks,” IEEE Trans. Consumer Electron., vol. 54, no. 3, pp. 1177-1184, Aug. 2008.

H. Uchiyama and E. Marchand, “Object detection and pose tracking for augmented reality: Recent approaches,” in Proc. 18th Korea-Japan Joint Workshop on Frontiers of Computer Vision, Feb. 2012.

A. M. Ullah, Md. R. Islam, S. F. Aktar, and S. K. A. Hossain, “Remotetouch: Augmented reality based marker tracking for smart home control,” in Proc. 15th International Conference on Computer and Information Technology, pp. 473-477, 2012.

.