An Implementation of Generic Augmented Reality in Mobile Devices

Qingfeng Zhang[1], Weilong Chu[1], Changhong Ji[2], Chengyuan Ke[3], Yamei Li[1]

1. Dep. CS, Jinan Univ. Guangzhou, P.R.C.
2. SANGFOR Inc., Shenzhen, P.R.C.
3. Gree Electric Appliances, Zhuhai, P.R.C.

***Abstract*—There are two problems in the existing mobile augmented reality (MAR) system, one is the difficulty in developing and the other is the lack of versatile AR observer. To address them, a generic MAR framework was proposed in the paper, which contains three components: a versatile observer which is run on smart mobile device to see the AR effect produced by MAR application, an MAR server that provides network and data service for the MAR application, and an MAR application customizer which is used by developers to tailor their desired applications. The Vuforia SDK is used to implement the observer, and the XML technology is applied to achieve the goal of customizing MAR application. The generic framework enables developers to do code-free development, and provides the convenience of observing different MAR applications by one MAR observer. The experimental results show that this framework reduces the difficulty and time in developing AR application and makes it easy to observing AR effects by users.**

Keywords—Mobile Augmented Reality; Augmented Reality Framework; Vuforia SDK; Augmented Reality Observer

I. INTRODUCTION

Augmented Reality (AR) refers to a technique originated form virtual reality technology, which is used to superimpose virtual information on real scenes, and provide powerful interactive services [1]. AR has the properties of combining real and virtual objects in a real environment, running interactively and in real time and registering (aligning) real and virtual objects with each other [2]. The AR technology can meet the user's needs about experience of virtual things in the real world; improve the user's interesting and interactivity. Therefore, it has been applied in many areas [3]. AR displays may be classified into three categories based on their position between the viewer and the real environment: head-worn, hand-held, and spatial [4]. Here, because of the wide application of handheld devices, we choose the mobile phones (or tablets) as the AR display.

In recent years, mobile devices have been increasing their power and features which make Mobile Augmented Reality (MAR) becoming possible. As a fresh experience, MAR is welcomed by users, and spreading through the mobile app markets. Now, it is the main direction of the future development of AR technology [5]. There exist many scenarios of MAR application. Shatte et al. put forward a MAR library management system based on context-aware [6]; Jin and Park achieved an interactive augmented reality system using vibro-tactile pad [7]; Newton et al. developed an MAR application for the Bukit Brown cemetery navigation [8]. Most of the existing MAR applications run typically on their own, with minimal or no linkage at all to centralized systems [9], having most of the content locally stored on the mobile device. That means different MAR applications need to install different software and resources, and each AR application can be used only for single case and purpose.

From the view of convenience, user expects that one MAR application is able to adapt to as many cases as possible. Hence, multipurpose MAR application is needed. Stephan Gammeter et al. created an MAR application which has a server-side object recognition and client-side object tracking [10]. As they proposed, the server stores the resources used for recognizing and be burdened with the computing of recognizing images, while the client tracks the target object and displays the AR effect. The system strengthens its flexibility by applying the CS architecture, and has a broader area of application than the traditional AR. Nonetheless, the system provides only the AR service of recognizing and annotating landmark images and videos. Petors et al. implement an AR system with C/S architecture which can add tags to the user's camera photos with the help of GPS positioning, and the photos of relevant tags from the famous photo-sharing site Flickr can be presented in the user's smart phone's camera [11]. The application's architecture is also flexible, but its service is still limited to specified purpose of augmenting photos with GPS information.

Beside an MAR applications usually developed for single purpose, its development is quite complex. Generally, it is difficult to make a useful MAR application because a variety of techniques have to be combined to use [12]. For the users' and developers' convenient, we have carried out some preliminary studies on MAR framework on the Android and iOS platform respectively [13, 14].

Based on the above, we propose and implement a framework which reduces the difficulty of developing MAR applications and has a relatively strong versatility. The framework includes an MAR application customizer with graphical user interface (GUI) whose function is registering target image and configuring AR effect, an MAR web server that is used to store and deliver resource files which is used to show MAR effect and an MAR observer by which user can watch the customized AR effects. Especially, the MAR application customizer is a simple MAR tool kit by which developers can easily develop some MARs instead of learning various technologies to do it.

978-1-4799-4419-4 /14/$31.00 ©2014 IEEE

The remainder of this paper is organized as follows: section II gives an overview over the framework and its implementation, sections III describes the key technologies needed when developing the framework, an example is showed in section IV, and section V is a conclusion.

II. GENERIC MAR APPLICATION FRAMEWORK

For MAR applications based on 2D images, we have designed and implemented a generic framework, which has the following features:

- A generic MAR observer in client-side is provided to browse various MAR application, user don't need install diverse observer for diverse application.
- When watching images by the observer, user can see a lot of augmented information including text, URL, panorama, video and 3D model, and operate them by corresponding way.
- When users browse AR effects in mobile device, only relevant resources are downloaded from server, and the requirements of local storage and computing power are lowered.
- A tool kit is supplied to easily tailor an MAR application based on images without writing code.
- A web server is set up to provide the AR service including storing the entire resources in specific folders for every MAR application and delivering relevant resources for every user.

The entire framework consists of three parts: MAR observer, MAR web server and MAR application customizer. MAR application developers will use the customizer to tailor their MAR application, and then upload recognition files (mentioned below), configuration files and resources files to the MAR server. MAR observer will download related resources from MAR server by HTTP or FTP, and display them by the predefined way. The overall framework is shown in Figure 1.

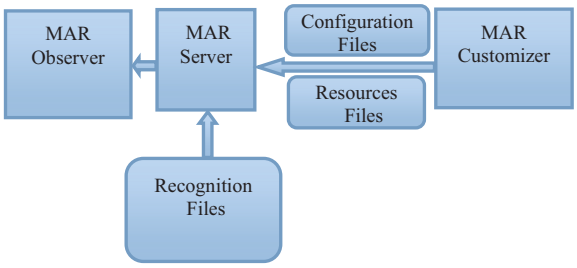

Fig. 1. The generic MAR framework

A. Vuforia SDK

Before detailing the various parts of the system, we need to understand the Vuforia SDK used in the framework. Vuforia is an Augmented Reality SDK based on artificial mark [15]. It contains stable image recognition and tracking technology that can recognize images and texts. After target image is uploaded and processed by the website of Vuforia, the user will obtain target image recognition files. The MAR application is able to recognize and track the target image as long as loading its recognition files in the MAR observer. Users do not need to consider the implementation of recognition and tracking image technology.

B. MAR application customizer

MAR application customizer is a tool kit for developing MAR application based on images. It allows user to define the interaction behaves between user and the target image processed by the website of Vuforia. Finally, an XML configuration files is generated by it.

In the customizer, we use hot points to link the region of interesting on target image and its relative resource. Hot point refers to a circular, a rectangular or a polygonal region with events properties where some resources will be shown or manipulated when event is triggered by user. Some actions happened in a hot point can trigger corresponding events. For example, a click in a hot point triggers a click event.

One hot point owns one or many events, and every event is associated with some resources and their operation, i. e. a click event may associated with the playing (an operation) a video (resource) in certain hot point.

The resources supported by our framework cover text, URL, video, panorama and 3D model. Every kind of resource corresponds to some basic operation. For example, the resource text has four operations: displaying, disappearing, zooming out and zooming in.

All hot points, events, resources, operations on one target image and their relation are recorded in a configuration file in XML format. Using the customizer, the developers can set hot points on the target image, define the events associated with hot points. Then the customizer will generate XML file that will be interpreted by the MAR observer.

Fig.2 shows a script of XML configuration file. A hot point with two events is configured in the snippet. The two events express two responses: opening relative website on click, rendering a 3D model on double click.

```
<targetConfiguration>
    <hotlink>
        <name>test</name>
        <type>pointHotlink</type>
        <point>-114,1</point>
        <event>
            <eventType>click</eventType>
            <resourceType>webpage</resouceType>
            <resourceLink>http://www.test.com</resouceLink>
        </event>
        <event>
            <eventType>dooubleClick</eventType>
            <resourceType>model</resouceType>
            <resourceLink>mymodel.3ds</resouceLink>
        </event>
    </hotlink>
</targetConfiguration>
```

Fig. 2. XML code snippet

C. MAR Server

MAR server is the core of the framework. We take the web server running Internet Information Services (IIS) as the MAR server, which includes the FTP service. It is a database in which a lot of data files are stored in the way of one folder for one application's entire resources. In addition, it is a bridge between MAR customizer and MAR observer. Resource is uploaded from the former and downloaded by the latter.

All kinds of data files stored in the MAR server can be divided into three parts:

- Image recognition files (.Dat and .Xml files) generated by the Vuforia website. It is used for recognizing and tracking the target image.
- Configuration XML files produced by MAR application customizer. It is used for locating and customizing the AR effect.
- Resource files which will be used for showing or rendering AR effects. They include 3D models' files, video files, images files, etc.

D. MAR Observer

MAR observer is application running on mobile smart device with iOS or Android operating system (such as mobile phones, tablet PCs, etc.). It identifies which MAR application is requiring data files and determines what data files will be downloaded from MAR server. After obtaining relative data files, it will make use of them to show the AR effect in the right position when relative event triggered by certain action.

From the MAR users' view, the framework is C/S architecture. The MAR observer is the client-side application which provides the ability of browsing AR effect for user, and MAR server is server-side application which offers HTTP and FTP service and data storage service. The ability of observing AR effect is reached by the observer's interpreting XML configuration files.

It is a key task to obtain the target image. How to know where are the target images? We use two dimensional codes to solve this problem. Each target images has a unique identifier, which contains the path of target image in the MAR server. The path can be used to find and download the resources files. For the convenience of the users, the unique identification is stored in two-dimensional code which was attached to the target image. When the MAR observer was used, the users only need to scan the two-dimensional code, and then files will be downloaded automatically. After the files is loaded, Vuforia rendering function calculates the coordinate and size of AR effect of target image based on the camera's current position, and then display the AR effect.

III. KEY TECHNOLOGIES

A. Fitting of the hot points and touch points

The regions on the target image configured to trigger events are called hot points, and the corresponding region on the screen is called touch points. The Vuforia does not provide directly method to convert the screen coordinates to the coordinates of the hot points, so how to accurately matching the hot points with touch points is an important problem. As we all know, different devices may have different resolutions. In the framework, firstly, we convert the coordinate of touch point into standard device coordinate, and then convert the different resolutions into a unified standard, and the standard device coordinate into coordinate of the viewpoint, and finally the viewpoint coordinate into coordinate of the hot point, through a series of transformation above, we will convert the coordinate of touch point to the hot point's.

B. 3D Models Showing

The AR observer needs to be run on the Android and the iOS platforms. However, iOS cannot directly read any 3D model, and OPENGL ES cannot display the 3D model directly. The scene data, material, mesh and other information of 3DS model used in the framework are stored in "blocks". We read the block information via the recursive method, and then import into OPENGL ES for loading and displaying [16].

C. AR Video Playing

The impressive feature of AR video is that it overlays in the target image. It makes users feel as if the video is playing directly on the image. The puzzle of implement the AR video comes from the device and its camera's movement causing that the position and angle to the target image changes frequently. Generally, video playback is processing on a view layer. When developing the system, we take each frame of AR video as view layer, and then render the frame as a texture on the target image. With the playing of video, the system changes the corresponding texture image so that the video content can be accurately superimposed on the target image.

IV. EXPERIMENT AND ANALYSIS

A customized MAR application is created as the following steps during which the framework proposed in the paper is made good use. Finally, the application was tested on an Android mobile device.

There is an original target image as shown in Fig. 3. The six green dots in the image will be configured to show different AR effect. In our test, our aim is to customize an application that is used to show a model when clicking "Point 1" and show an AR video when double click "Point 6".

Fig. 3. Original target image

Firstly, the target image is uploaded into the Vuforia website, and then the two files (.xml file and .dat file) are

downloaded and put into MAR server. Secondly, the positions of the hot points are configured using MAR customizer, and their corresponding resources, and the type of events are also added and linked. Fig. 4 shows the GUI for adding and linking hot points and their resources. After these steps, the customizer will generate an XML configuration file. Then these files mentioned above are uploaded to the MAR server.

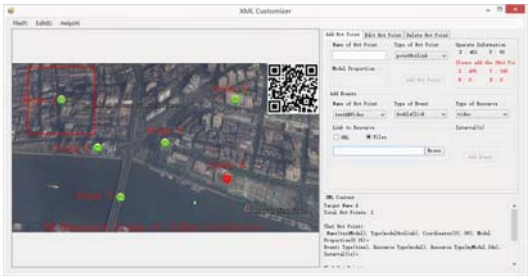

Fig. 4. Customize the target image

After uploading the file to the MAR server, the user can start the MAR observer in their mobile device. When the phone's camera is facing the image as Fig. 3, the application will download the files which are just uploaded to MAR server through the internet. When finishing the downloading task, the MAR application will automatically use these files. Thus, the AR effect would be observed on the screen. After the user click the hot point "Point 1", there will be a 3D model, and after the user double click "Point 6", an AR video will be displayed on the target image. The Fig. 5 shows the result.

Fig. 5. Rendering 3D model and AR video

Through the result of our test, we can find that there is no need to program and learn lots of knowledge when using our framework. Creating a customized MAR application is very fast and simple. The developer just upload their own target image and configure the AR effect with help of MAR customizer, and an MAR application can be created. Our MAR observer is versatile and universal because it can observe any MAR application produced by our customizer regardless of the tailored application's purpose. Hence, the framework is very efficient.

V. CONCLUSION

We put forward a generic framework which includes three parts: MAR application customizer, MAR server and MAR observer. The customizer is a quickly developing tool kit for MAR application based 2D images. The MAR server offers the data service for observer and customizer. The MAR observer is a versatile app in mobile device that can browse any AR effect from the any MAR app produced by our customizer. The framework significantly improves the efficiency of the development, and enhances the observers' versatility. Experimental result shows the framework's effectiveness.

ACKNOWLEDGMENT

This work is partly supported by the National Natural Science Foundation of China (Grant No: 11273014), by the Science and Technology Innovation Funds for Universities of Department of education of Guangdong Province (Grant No. 2013KJCX0020) and by the Fundamental Research Funds for the Central Universities.

REFERENCES

[1] Zhao Xinping. Survey on Virtual Reality. Science in China (Series F: Information Sciences). 2009, 39 (1):2-46.

[2] Ronald Azuma,Yohan Baillot,Reinhold Behringer,Steven Feiner,Simon Julier,Blair MacIntyre. Recent advances in augmented reality, IEEE Computer Graphics and Applications, 21(6) (2001): 34-47.

[3] Ronald T. Azuma. A Survey of Augmented Reality. Tele-operators and Virtual Environments 6(4) (August 1997), 355-385.

[4] D.W.F. van Krevelen and R. Poelman. A survey of augmented reality technologies, applications and limitations. International Journal of Virtual Reality, 2010, 9(2): 1-19.

[5] Marco de Sa, Elizabeth F. Churchill,Katherine Isbister. Mobile augmented reality: design issues and opportunities. Proceedings of the 13th International Conference on Human Computer Interaction with Mobile Devices and Services. ACM, 2011: 749-752.

[6] Shatte A, Holdsworth J, Lee I. Mobile augmented reality based context-aware library management system. Expert Systems with Applications, 2014, 41(5): 2174-2185.

[7] Jin M S, Park J I. Interactive Mobile Augmented Reality system using a vibro-tactile pad. VR Innovation (ISVRI), 2011 IEEE International Symposium on. IEEE, 2011: 329-330.

[8] Newton O N, Deshan C, Pang N, Nakatus R. Mobile augmented reality for Bukit Brown cemetery navigation, Proceedings of the 11th ACM SIGGRAPH International Conference on Virtual-Reality Continuum and its Applications in Industry. VRCAI'12, 2012: 59-62.

[9] Xun Luo. From Augmented Reality to Augmented Computing: A Look at Cloud-Mobile Convergence, 2009 International Symposium on Ubiquitous Virtual Reality (ISUVR 2009), 2009, pp. 29-32

[10] Stephan Gammeter, Alexander Gassmann, Lukas Bossard. Server-side object recognition and client-side object tracking for mobile augmented reality. 2010 IEEE Computer Vision and Pattern Recognition Workshops (CVPRW), 2010:1-8.

[11] Petors Belimpasakis, Petri Selonen, Yu You. Bringing User-Generated Content from Internet Services to Mobile Augmented Reality Clients. Virtual Reality Workshop (CMCVR), 2010 Cloud-Mobile Convergence for. IEEE 2010:14-17.

[12] Zhou Guo-zhong. Technology and Application of Mobile Augmented Reality. Geomatics & Spatial Information Technology, 2012, 35(9): 140 -144.

[13] Ke Cheng-yuan. Design and Implementation of an Augmented Reality Framework System Based on the Android. Guangzhou: Jinan University, 2013.

[14] Ji Chang-hong. An Augmented Reality Universal System on iOS Platform. Guangzhou: Jinan University, 2013.

[15] Zha Jun-yuan, Qin Wen-hu, Zhao Zheng-xu.Study of augmented reality based on artificial mark. Computer Engineering and Design, 2008, 29(18) : 4819-4821.

[16] Chen Xue-Mei, Xie Qing-Zhong. Superficial Analysis on a method of Importing and displaying 3DS Models Using OpenGL. Journal of Jilin Provincial Institute Education (Mid-Month), 2012, (11):86-87.