An Augmented Reality System Using Qr Code as Marker in Android Smartphone

Conference Paper · May 2012

DOI: 10.1109/SCET.2012.6342109

2 authors, including:

Hong Jeong
POSTECH POHANG UNIVERSITY OF SCIENCE AND TECHNOLOGY
138 PUBLICATIONS **583** CITATIONS

SEE PROFILE

An Augmented Reality System Using QR Code as Marker in Android Smartphone

Kong Ruan
Pohang University of Science and Technology
Pohang, Korea
e-mail: ruankong@postech.ac.kr

Hong Jeong
Pohang University of Science and Technology
Pohang, Korea
e-mail: hjeong@postech.ac.kr

***Abstract*—In this paper, we present an Augmented Reality (AR) system using Quick Responsible (QR) code as the marker for Android Smartphone. QR code has many advantages to be a marker. It can encode relatively larger amount of marker information in an easy and standard way, also it has the capability of error correction. Basically the system detects the marker, decodes its information and overlays a 3D object on the marker. As QR code is widely used today, our idea of combining QR code and AR to develop an application in handheld smart device can extends to many fields.**

***Index Terms*—AR QR code Android OpenCV OpenGL**

I. INTRODUCTION

Augmented Reality(AR) is a concept to integrating virtual information to a real-world environment. According to different tracking technology, AR systems can be grouped as two general types. One is Geographical Information System (GIS) using GPS data and solid state compasses; the other one is using image recognition.[1] GIS-based AR system needs accurate location information. However, so far, even though GPS data is able to handle the range of meters, it's not good enough in many cases. Using image recognition technology to develop AR system is very popular these days. This vision-based system also has two sub types, maker-based AR and maker-less-based AR.[2] Without doubt maker-based AR is more mature, but also it needs more computation and memory.

For maker-based system, usually a special pattern is put into the scene as the "maker". Some previous interfaces, such as ARToolkit, have their own makers with a fixed format. (Fig. 1) This makes them easier to be tracked in a way, but to generate a unique marker of user's own needs accurate measurement and also the encoded information is limited and nonstandard. Recently Quick Response (QR) code is a popular way for encoding information due to its fast readability and comparatively large capacity. These advantages also make QR code a good marker. And since it is widely used in many fields, we may extend our application to different areas with different usages.

Lately Mobile Augmented Reality (MAR) has made a great progress thanks to the development of some handheld smart devices, such as Smartphone. Smartphone with integration of camera, general-purpose processors and sensors is a suitable platform for AR. In our proposed system, we used QR code as our marker and planted the application to Android Smartphone.

II. MATERIALS AND METHODS

Figure 1. ARToolKit Marker

Similar to the mechanism of the applications developed using ARToolkit, our system also has three steps: detecting the marker; decoding the marker and drawing virtual objects. But since our marker is QR code, not the same pattern format in ARToolkit, the implementation is quite different from ARToolkit in detecting and decoding.

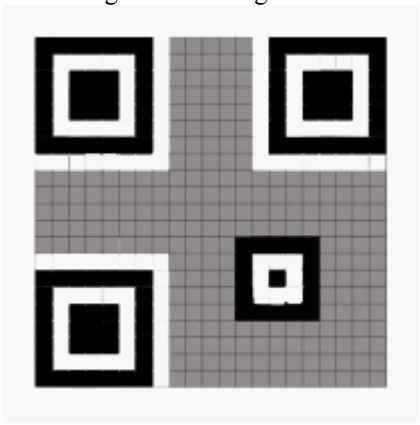

Figure 2. Finder Patterns and Alignment Pattern in QR Code

The first step is detecting the marker. All QR codes (Fig. 2) have three position detection patterns, named finder patterns, which are placed in the top left, top right, and bottom left corners. Some QR codes also have another kind of patterns named alignment pattern. The number of alignment pattern varies in different version. But we only focus on the one located in the bottom right corners. Finding these four patterns is the key procedure in the first step. For the three finder patterns, each time we kept horizontally tracking the "black/white/black/white/black" modules in 1:1:3:1:1 ratio, counted the center of the modules as one potential finder pattern, and finally scanned down vertically through the center of the suspected finder pattern to see if the same proportion is detected. For the alignment pattern, we simply did almost the

978-1-4577-1964-6/12/$26.00 ©2011 IEEE

same thing with the finder pat-tern, only the ratio turned out to be 1:1:1:1:1. In QR code, these four patterns are located in their fixed position. Depend on that point, we used four source points and four destination points to calculate the perspective transform matrix implied between them. However, not all QR codes have the alignment pattern. In that case, we made up a bottom-right point (bottomRightX, bottomRightY) using the information of the finder patterns.

$$bottomRightX = (topRight.getX() - topLeft.getX()) + bottomLeft.getX() \quad (1)$$

$$bottomRightY = (topRight.getY() - topLeft.getY()) + bottomLeft.getY() \quad (2)$$

Then we called OpenCV calibration functions to calculate the camera's extrinsic parameters including rotation vector and translation vector. Since we would use OpenGL to draw virtual object in the third step, these vector data would help to form an OpenGL transformation matrix.

The next step is decoding the marker. The basic framework for reading 2d code is four parts: regions of interest (ROI) detection; code location; code segmentation and data decoding.[3] In our system the first two procedures has already been done in the first step, detecting the marker part. The ROI is generated by an image tessellation with a fixed size block. The detailed operation is first computing the gradient magnitude (g) of the points in each block, then selecting the points that g is greater than threshold G and finally evaluating the directional histogram. In the case of QR code, the histogram will be two orthogonal modes. Code location is to detect the standard edge symbols. The symbol of QR code is the finder patterns and alignment patterns. For the next procedure, code segmentation is to extract the boundary of the code. QR code uses interleaved data blocks to encode information. That means while encoding, one byte of data uses 8 units blocks in code area, but each byte has different shapes since there are other patterns need their fixed position. (Fig. 3) So now we segmented the code area to byte data. After that, we finally decoded each byte to text information.

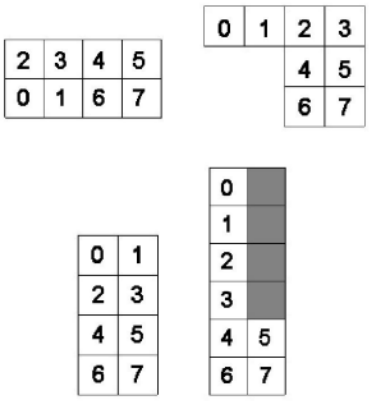

Figure 3. Different Segmented Blocks

The final step is drawing virtual objects. In our system, we had two kinds of markers and they represented the object information of cube and pyramid separately. (Fig.4)As we mentioned before, we used OpenGL to do the drawing. In the first step, we had obtained the transformation matrix. In the second step, we had obtained the object information. So here we could use the transformation matrix as the parameter to call function gl.glTranslatef and gl.glRotatef in OpenGL to get the exact location, and finally draw the right object on the marker.

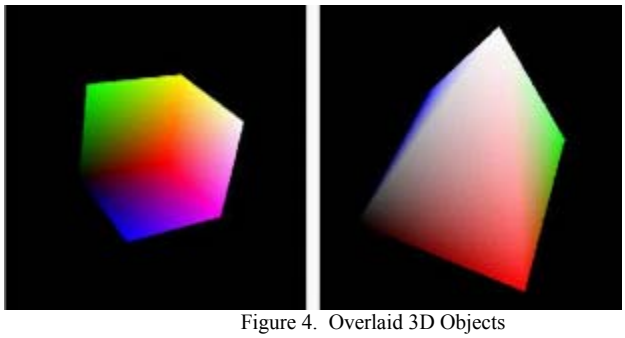

Figure 4. Overlaid 3D Objects

III. RESULTS

We planted our system to an Android phone(Samsung Anycall nexus/s SHW-M200S Target Android 2.3.3) to check the performance. We generated two QR Code (Fig. 5)

Figure 5. Test QR Code

with text information “cube” and “pyramid” in a website (http://qrcode.kaywa.com/). These two test QR code don’t have alignment patterns. So first 3 finder patterns were detected and the coordinate of the center point was calculated. These four points were marked as green dots. Then the de-coded information was shown at the bottom of the screen. And finally the corresponding 3D object was overlaid on the marker. (Fig. 6)

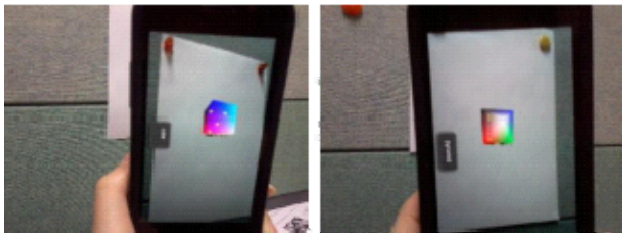

Figure 6. Result

IV. DISCUSSION

Compared with ARToolKit, our system using QR Code as a marker has some advantages. First as it is called quick response code, it can be scanned, read and decoded very fast, so the processing time is shorter. Also QR code can be generated easily in a standard way and the encoded information amount is relatively large. What’s more important, QR code has one good feature, Error correction. That means even if the code is dirty or

damaged within the acceptable range, QR code can still be decoded successfully. We covered part of one test maker for several times and checked out the result with ARToolkit. Since the maker's error correction capability is level L, so 7% of code words can be restored. But ARToolkit couldn't decode the marker if it's partly covered.

V. CONCLUSION

We designed and implemented an AR system using QR code as the marker for Android Smartphone. The system had three major procedures to go: firstly in the camera captured view, detect the location of QR code and calculate the transform matrix between the device and that QR code marker; secondly decode the QR code and get the marker's information; finally draw the specified virtual 3d object on the QR code marker. Comparing to the marker pattern used in ARToolKit, QR code is generated in a standardized and efficient way while encoding marker information. Now QR code usage is expanding in various fields, using Smartphone to scan QR code is becoming a popular and common way to get information, so the research topic on combining QR code and AR is very promising.

REFERENCE

[1] Haruhisa Kato,Tsuneo Kato，A Marker-Less Augmented Reality Based on Fast Fingertip Detection for Smart Phones，2011 IEEE International Conference on Consumer Electronics (ICCE)

[2] Hanhoon Park , Jong-Il Park, Invisible Marker Tracking for AR - Proceedings of the Third IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR 2004)

[3] Ouaviani. E, Pavan. A, Bottazzi. M, Brunelli. E, Caselli. F, Guerrero, M.; , "A common image processing framework for 2D barcode reading," *Image Processing and Its Applications, 1999.*