2011 International Conference on Mechatronic Science, Electric Engineering and Computer
August 19-22, 2011, Jilin, China

Research and analysis of the Virtual Reality with FLARToolKit

Yongxin Yan
International School
Beijing University of Posts and Telecommunications
Beijing, China
e-mail: ee08b044@bupt.edu.cn

Xiaolei Zhang
International School
Beijing University of Posts and Telecommunications
Beijing, China
e-mail: ee08b226@bupt.edu.cn

***Abstract*—Currently most technologies used by virtual studio systems are extended applications of Virtual Reality (VR) technologies. But for small TV stations and web stations, virtual studio based on traditional VR is so expensive while the FLARToolKit technology used by Augmented Reality (AR) technologies is their top option. However, the calculation speed of FLARToolKit is a little slow and the frame rate is very difficult to reach the live broadcasting video level. My improvements are to apply adaptive thresholds, use sample blurring and change identification frequency, which solves the shortcomings and makes the Virtual Studio faster and better.**

Keywords-FLARToolKit,Augmented Reality (AR), adaptive thresholds, sample blurring

I. INTRODUCTION

Virtual Studio is a newly developed TV program making technology in recent years, as a combination of traditional studio's Chroma Key technology and virtual reality technology. It perfectly combines the real video shot by cameras and virtual scenes made by computers to create a fantastic virtual world (Shown in Fig1). This technology frees the TV program makers from the restrictions on time, space and props production, allowing their wildest imagination, which greatly improves the TV station's program creation and production capacity.

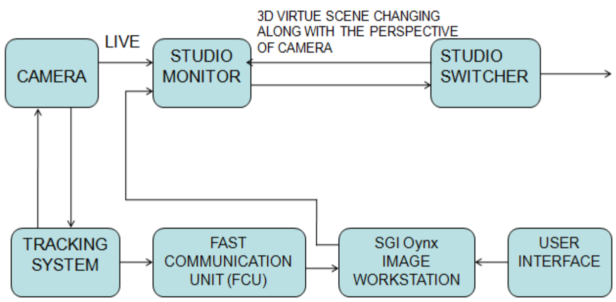

Figure 1. Diagram of RT-SET Larus virtual studio system [1]

Most technologies used by virtual studios are extended applications of VR technologies. To some extent, the VR technologies use computers to make virtual scenes (including foreground and background), and the presenter host the program in the virtual world. It requires people to know how to judge the relative relationships of positions of cameras, presenters, and the virtual background from computers, in order to synchronize them. The key to synchronize is continuously tracking and obtaining the motion parameters of the cameras, including the lens motion parameters (e.g., zoom, focus, aperture), head movement parameters (e.g., pan, tilt), and spatial parameters (X and Y positions from the ground, and height Z), etc [2]. Currently mature tracking technologies include sensor-based camera tracking systems and pattern recognition systems. The former needs to put on complicated sensors and encoders, and the positioning and calibration of cameras are very complex, while the latter must use hardware digital video processor for pattern recognition. Either technology needs extremely high procurement cost, so it is impossible for small TV stations or Internet broadcasting stations to build this kind of high-cost virtual studios [3].

II. AUGMENTED REALITY

Augmented Reality, also known as Mixed Reality, is a hot research topic in many foreign famous universities and research institutes, which applies virtual information upon the real world via computing technologies, to superimpose virtual objects upon the real environment in real time into the same picture or space. AR provides information different from what humans can perceive in general conditions. It not only demonstrates the real world information, but also displays virtual information at the same time to complement each other. AR technologies can be widely applied, e.g., research and development of sophisticated weaponry and aircrafts, data model visualization, virtual training, entertainment and art, etc.

Different from VR, AR technologies identify the objects in the real world, and generate virtual objects on the real object to enhance the images. Since AR's core technology is the algorithm for the processor to identify real-world images, which is basically software development either the development of its implementation platform or the algorithm optimization, therefore the huge hardware procurement cost is saved. Besides, compared with the VR technology's complex hardware sensors, powerful processors, and complex delay control system, AR demands much less on processors [4]. Therefore complex delay processing can be ignored and there is no need for lots of sensors, at the same time the cameras can move freely within a certain range.

For the audiences, they won't care what technologies are used by the studio, but only care about the final effect of the pictures. Even though the complex virtual studio using VR technologies for virtual images should be better than the effect provided by AR rendering partial images. But for small TV stations and web broadcasting stations, in order to bring different visual enjoyment, the expensive virtual studio based on traditional VR technologies is not their top option, but the relatively simple and cheap AR technologies can be helpful.

III. FLARTOOLKIT

However, AR can't be applied to low-level users until the November of 2008, the programmer Tomohiko and his team from Japan released their code NyARToolKit (Java version) and FLARToolKit (ActionScript3.0 version) as open source on the Internet. [5]

FLARToolKit needs support from 3 aspects to carry out basic identification and capture functions.

- Camera parameter file, which records the camera parameters and configuration information using binary code
- Marker/pattern: FLARToolKit will analyze its graphic information and compare with the target area in the real environment. The pattern is by default composed by black borders and internal images.
- Detector, with the following functions:
  a) *Obtaining real-world image information from cameras:* completed by the camera detectors inside the FLASH, which is then added to the stage display objects (Shown in Fig2)

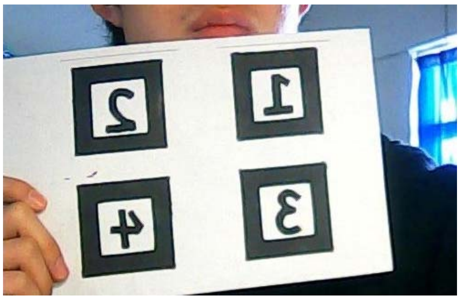

Figure 2. Real-world image

  b) *Converting the pixel image to binary image:* FLARToolKit will convert the display image information captured from cameras into binary images (with each pixel black or white). Whether a pixel is converted to be black or white is determined by a "threshold", where the pixel's brightness level is higher than the threshold, FLARToolKit converts it to black, and white if not. (Shown in Fig3)

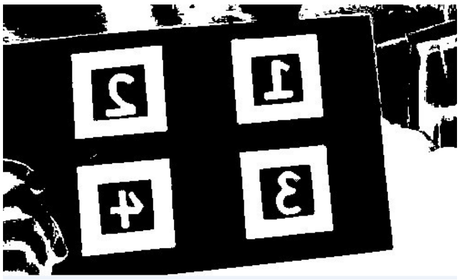

Figure 3. Binary image

  c) *Labeling and finding squares:* the FLARToolKit functions BitmapData.getColorBoundsRect and BitmapData.floodFill mark black and white pixels close to each other into groups (with each area in different colors). Because a marker is a square, FLARToolKit will look for squares in marked areas as candidate zones. (Shown in Fig4)

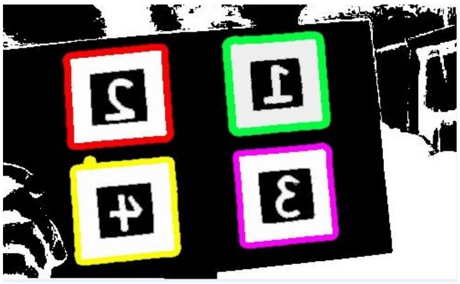

Figure 4. Square image

  d) *Comparing the candidate zone with patterns:* it compares the central 50% of the candidate zone with the zones defined in the pattern file, calculating each similarity by comparing graphic geometric vertexes and selecting the highest. If the highest similarity is greater than a given minimum similarity "threshold", then return this zone as matched. [6]

IV. IMPROVEMENT AND EVALUATION

After understanding FLARToolKit, I carried out initial debugging and found it not only complex and difficult to grasp, but also slow in calculation speed. If it is applied in live video broadcasting, the frame rate is very difficult to reach the broadcasting level. So I have made the following improvements.

A. From color identification markers to adaptive thresholds

As mentioned above, FLARToolKit identifies the black pixels in real images, by filtering the black areas with the class "FLARSingleMarkerDetector". But on the one hand, there are many black objects in the real world; while on the other hand, captured images by most cameras are dim, which makes FLARToolKit to conclude wrongly with many more candidate zones than there should be because it depends on the brightness to filter. As the similarity calculation is applied for each candidate zone, the calculation load is obviously increased unnecessarily.

In the beginning, I decided to improve FLARToolKit for it to recognize colors, e.g. filtering red zones (as there are relatively fewer red objects in real life). But colorful markers are affected by ambient light to a great extent, and the same color has different brightness in different lighting conditions, which requires the identification to be very precise and extremely strict for ambient lighting conditions.

Finally I decided to use variable thresholds to solve the binary image conversion speed problem, using the class "adaptiveThresholding" of the open frame "FLARManager" to modify the thresholds automatically. I set the variable "adaptiveThresholdingSpeed" (which determines the calculation speed of thresholds) to a relatively big value "0.5" (which enables fast calculation of new threshold value suitable for the environment, but relatively slow in changing threshold values). Because the virtual studio application generally provides stable ambient brightness, I set the variable "adaptiveThresholdingBias" (threshold bias) at "-0.1" to be able to work in dim or dark conditions. (Shown in Fig5)

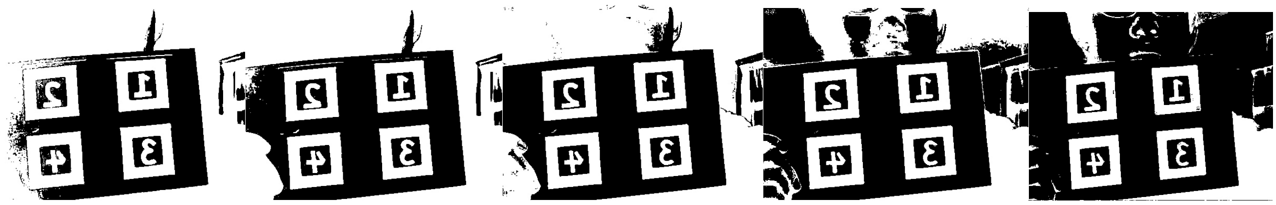

Figure 5. The size of black zones to identify is different with different threshold values set. The larger the black zone is, the heavier the calculation load and slower identification will be

B. Sample blurring

As mentioned above, FLARToolKit treats all matching squares as candidate zones, comparing them with markers, and calculates similarity. As our marker is a rigid rectangle, I can blur the image intentionally to minimize the rectangles for the computer to identify in order to speed up the calculation. By modifying the value of "sampleBlurring", I can get the optimized value while guaranteeing the image identification results. Of course, the computer will then not be able to identify our marker in when the reversion is significant. (Shown in Fig6)

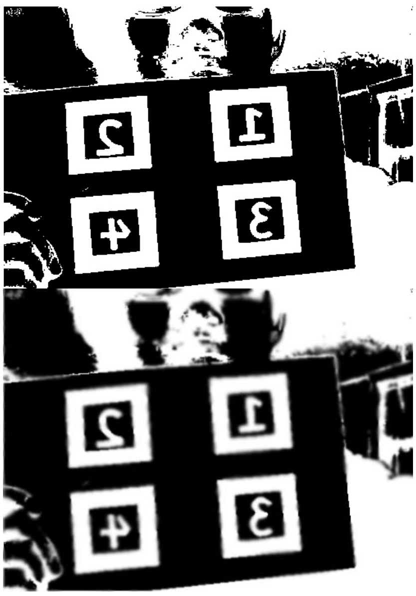

Figure 6. Binary images before and after blurring.

C. Changing identification frequency

In the beginning when I debugged the code, I found images pausing regularly. After exploration and learning, I found that FLARToolKit made each identified image into a bitmap image to form video signals, and when the identified frame rate is very high, the new data was sent to CPU before the computer processed the data previously received in a short time, which caused the "pausing" error and is definitely not allowed in virtual studio. Then I added the method "MARKER_REMOVED", and modified FLARToolKit's time gap between 2 identifications, which lowered the identification frequency but increased calculation speed.

By the comparison between the before and the improved, the average time of improved technology used is 73.6% of the time that former FLARToolKit technology, it means the improvement is basically successful and the result of the frame rate is near to 23 fps, closer to the live broadcasting video level 25 in China.

V. CONCLUSION

For the FLARToolKit technology being used by Augmented Reality technology for the low-level customers, by deeply analyzing and studying, I propose three effective methods to improve it — applying adaptive thresholds, using sample blurring and changing identification frequency.

By multiple measurements, it is proved that the improvement really can save time as I have analyzed before, basically solves the shortcomings. The virtual studio system based on AR technologies has initially verified the feasibility of using AR to replace VR, using software instead of hardware systems, for the application of virtual studio in the broadcasting industry. Following this mindset, some small TV stations and web TV stations can create their own "semi-virtual studio", using augmented reality technologies to offer the audience with refreshing visual enjoyment.

REFERENCES

[1] Gibbs.S, Arapis.C, Breiteneder.C, and Lalioti.V, "Virtual studios: an overview", Multimedia, IEEE, vol.5(1), Aug.2002.pp18-35, doi: 10.1109/93.664740

[2] Luo Deyuan, Qin Dongxing, and Huo Yaqin. "Reliability of the parameters calibrating algorithm of the virtual studio camera",

Intelligent Mechatronics and Automation(IMA), IEEE Press, Feb. 2005, p905-908,doi:10.1109/ICIMA.2004.1384328

[3] Po Yang, Wenyan Wu, Moniri, M, and Chibelushi, C.C. “RFID tag infrastructures for camera tracking in virtual studio environment”, Visual Media Production, IEEE Press, Feb.2008, p1-8

[4] Seoksoo Kim, “Virtual studio system for augmented reality & Chroma key processing”, 13th International Conference on Advanced Communication Technology (ICACT), IEEE Press, Feb.2011, p762-765

[5] Chao Chen, Jiean Zhang, “Design and realization of Computer Aided Instruction platform based on Augmented Reality”, 2010 3rd IEEE International Conference on Computer Science and Information Technology (ICCSIT) , IEEE Press ,July .2010, p515-518.

[6] H. Zongyi, "Designing and implementing outdoor augmented reality system based on ARToolKit, " Proc. Geoinformatics 2007: Geospatial Information Technology and Applications, May 25, 2007 - May 27, 2007, SPIE, 2007, pp204-208.