Designing Real-Time Vision Based Augmented Reality Environments for 3D Collaborative Applications

Peiran Liu, Nicolas D. Georganas
Multimedia Communication Research Laboratory
University of Ottawa
{peiran, georganas }@mcrlab.uottawa.ca

Pierre Boulanger
University of Alberta
pierreb@cs.ualberta.ca

Abstract

Augmented Reality (AR) is a variation of virtual reality. It allows the user to see computer generated virtual objects superimposed upon the real world through the use of some kind of see-through head-mounted display. Human users of such system can interact with the virtual world and have additional information, such as character description of physical objects and instruction for performing physical tasks in form of annotation, speech instruction, image, and 3D model.This paper describes our work of building a wireless augmented reality prototype, which supports video-based 3D graphics and a keyboard interface over wearable computers to interact with virtual objects. A new technique for identifying real world objects and estimating their coordinate systems is introduced. The method utilizes a Binary Square Marker, which can identify a great number of real world objects with markers tagged on them by using computer vision techniques.

Keywords: *Augmented reality; marker detection; collaborative application.*

1. Introduction

1.1 Background and motivation

From the human's perspective, the real world is composed of the physical materials that people can feel by their own senses. The information people get from their senses is very limited in some circumstances and extra information augmenting the real world could overcome the limitations. For example, when a repairman wants to fix a pipeline in the wall, a virtual map of the pipeline, overlaid on the real scene of the wall, will save the repairman a lot of time to find the broken pipeline. Another example: without a tour guide in an art museum, information such as the background of the artist or the music of his/her time will help the visitors to understand the work of art.

Augmented Reality (AR) is used to describe a system, which enhances the real world by superimposing computer generated information on top of it. AR is a variation of Virtual Reality (VR). VR technologies completely immerse a user inside a synthetic environment. While immersed, the user could not see the real world around him. In contrast, AR allows the user to see the real world, with computer-generated information or virtual information superimposed upon or composed with the real world. Therefore, AR supplements reality, rather than completely replacing it. Combining virtual information with the real world in 3-D space is useful in that it enhances a user's perception of and interaction with the real world. In addition, the virtual information, such as annotations, speech instructions, images, video, and 3D models, helps the user perform real world tasks.

1.2 Existing problems in AR

In general, AR is a system that combines real and virtual, is interactive in real time, and is registered in 3D [1]. Two basic technologies are available to accomplish the combining of real and virtual. One is using head-mounted display (HMD) equipment with a wearable computer, the other is overlaying computer-generated graphic images on the camera captured video using the monitor.

Proceedings of the 2002 IEEE Canadian Conference on Electrical & Computer Engineering
0-7803-7514-9/02/$17.00 © 2002 IEEE

The advantage of using a HMD is its mobility. When a user wearing a HMD and a wearable computer is moving around in the real world, the pose of the user's head is tracked in real time by the camera of the HMD. The object recognition and the graphic image rendering are accomplished by the wearable computer. That technology is most useful in outdoor applications where a computer graphic workstation is non-reachable. For AR applications, the reasonable image generation rate is 10 frames per second at least. According to a research by Durlach [11], delays greater than 60ms between head motion and virtual feedback impair the adaptation and the illusion of presence. The processing speed and graphic capability of the existing wearable computer products on the market are very limited.

Accurate dynamic registration is a key issue for an AR system. Magnetic and ultrasonic trackers used in VR are constrained by their inaccuracy and limited volume. Therefore, they do not provide accurate and portable registration in AR. Vision-based AR systems are those using video images captured by the camera to track the camera's position and orientation relative to the objects in the real world. The camera tracking has been extensively investigated in the field of computer vision that has the potential to provide the accurate registration data needed by AR systems. In computer vision, there are several ways for identifying objects in scenes. (1) Edge detection, which is applicable to the non-real-time applications due to the complexity of the algorithms. (2) Reconstruction of the 3D coordinates of a number of points in a scene given several images obtained by cameras of known relative positions and orientations, which is applicable to non-mobile applications. (3) Reconstruction of 3D coordinates of objects by detecting fiducial points or landmarks in the image obtained by one camera from a single view, which is applicable for AR systems due to its simplicity and low cost.

1.3 Related work

Before 1998, all of real-time vision-based AR systems implemented registration based on landmark detection techniques; these include boundary detection, color segmentation, watershed, and template matching techniques. Landmarks could be solid or concentric circular fiducials, ping-pong balls, LED, or the corners of a big rectangle [7][13][14]. Those landmarks might facilitate light-weighed fiducial detection and accurate 3D coordinate reconstruction, but what kind of information, or more precisely, what 3D model, annotation or texture image should be superimposed on the reconstructed object in the real world? In all those systems, the computers known in advance what the object with reference to the landmarks is and which 3D virtual model or texture image will be registered with the object seamlessly or what annotation will be displayed on the screen. This constraint hindered the vision-based AR technology to be used in more practical and large scale applications, in which the environment around the user changes dynamically and more than one object in the user's view need to be detected and augmented with virtual information. This raises a question. How to distinguish one real object from another based on the landmark detection techniques? In 1998, Jun Rekimoto proposed a method using a 2-D matrix code marker as landmark in his vision-based system [5]. The makers are 2-D barcodes, which can identify a large number of objects by unique IDs. The complexity of detecting the 2-D barcode and extracting bar code ID depends on how the 2-D matrix code marker is defined.

1.4 Technical contribution

We propose a Binary Square Marker recognition technique, which adds the error code detection and correction, and marker orientation functions into the 2-D matrix code. This technique makes the code of the new marker being recognized with higher probability. In this paper, a comparison among known marker techniques those of which are involved in AR applications is made.

To meet the need of 10fps image generation rate, and to exploit the mobility of the wearable computer and HMD, we moved most of the workload to a remote workstation and left only the image compression/decompression, display and data transmission to the wearable computer. The wearable computer (Xybernaut MA IV wearable computer with a 233MHz Pentium MMX) and the remote graphic workstation are connected by 10Mbps wireless Ethernet. This method is particularly effective in applications where multiple users collaborate with each other through a wireless-LAN. The remote workstation could work as a server to distribute data to wearable computers.

Although the AR technology has been developed since 1993, and many related researches have been done in the laboratory, only a few of commercial products using AR technology appear in the market today, such as Instructional System for Boeing Company, and the telecommunications services product for Bell Canada field-service technicians. In this paper, we introduce a real-time vision-based AR prototype using the enriched Binary Square Marker. We also propose several potential applications using AR and marker recognition techniques. Those applications could be used in industry, education and entertainment.

2. Camera tracking

Camera calibration is the process of estimating the intrinsic and extrinsic parameters of a camera, which are used to determine the relationship between what appears on the image plane and where it is located in the 3D world. Registration is the process of acquiring the objects' pose. The camera tracking can be regarded as a continuous update of extrinsic parameters. Camera tracking addresses the problem of accurately tracking the 3D motion of the camera in a known 3D environment and dynamically estimating the 3-D camera location.

Vision-based tracking approaches are based on detecting and tracking certain features in images. These features could be lines, corners, or circle points, which can be easily and unambiguously detected in the images and can be associated with objects in the 3D world. Our approach uses the corners of the square markers attached to moving objects for tracking. The overall work flow is illustrated in Fig. 1. Our implementation consists of four parts.

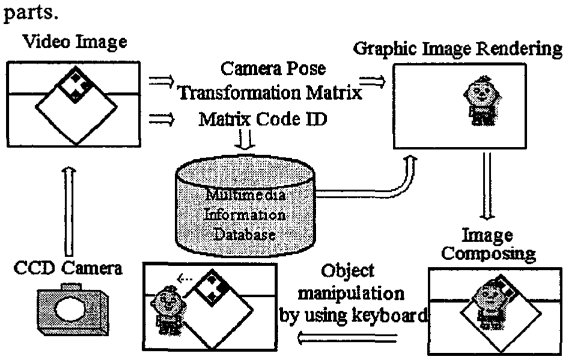


Fig. 1. Augmented reality system workflow.

2.1 Image preprocessing

(1) Binarizing the image.

The program uses a predefined threshold to binarize the video image. Binary images contain only the important information, and often can be processed very rapidly.

(2) Selecting the quadrilateral regions. These regions become candidates for the square marker.

(3) Searching and recording the four corners of the quadrilateral region found in step (2).

2.2 Pattern recognition

(1) The system recognizes the 2D binary code pattern inside the square marker region.

(2) Extracting the binary code.

Our algorithm is based on the following observation: Given the four vertices of the marker region, the projection of every point on the marker can be computed as a linear combination of the projections of the four vertices. The reference frame is defined as a non-Euclidean, affine frame. In our system, the marker pattern is defined as a 5x5 matrix. Every grid in the matrix represents one bit of the binary code of the marker. The whole pattern is in black and white color. The grid in black represents 1, and the grid in white represents 0 (Fig. 2). According to the theorem in [9], the division ratio is affine invariant, which means

$$(ABC) = (T(A)T(B)T(C)).$$

If A, B and C are three different arbitrarily chosen collinear points of the plane (or space) and T is an arbitrary affine transformation, (ABC) is the division ratio of the collinear three points A, B and C.

Clearly, since the relative location of every grid in the pattern is predefined, it is easy to calculate the central point coordinates of the grid in the projective plane with reference to the projections of four vertices of the marker. Consequently, the color or value of each grid in the pattern is determined by the color of the corresponding pixels in the binarized image.

(3) Error detection.

The pattern of a marker represents a 25 bits binary code. 12 bits are used to represent the marker ID. 4 bits determine the orientation of the marker in the camera's dynamic view. 8 bits are for the block sum checking, in which 4 bits are odd parity for rows and 4 bits are even parity for columns. In our marker pattern, the block is of 4 rows and 4 columns. The block sum checking can correct any single 2bit error in the bit block. The error bit is detected by checking the 8-bit parity code.

(4) Error correction.

If a code error is detected, the system can either drop the current image and do the image preprocessing again on the next image from step (1), or correct the error bits up to 2.

(5) Determine the orientation of the marker.

Four bits at the corners of the pattern determine the orientation of the marker in the camera's view dynamically. The system can keep tracking the orientation of the square marker, register the virtual model on the marker even if the marker rotate, and read the binary code of the marker in correct order.

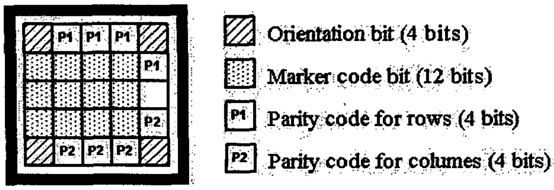


Fig. 2. Binary square marker.

2.3 Camera pose estimation

The recognized marker region is used for estimating the camera position and orientation relative to the marker tagged object. From the coordinates of four corners of the marker region on the projective image plane, a matrix representing the translation and rotation of the camera in the real world coordinate system can be calculated. Several algorithms have been created and implemented in experimental augmented reality systems. They are Hung-Yeh-Harwood pose estimation algorithm [2] and Rekimoto 3-D position reconstruction algorithm [6].

We currently use the algorithm proposed by Kato-Billinghurst-Weghorst-Furness [3], which incorporates lens distortion. Once the transformation matrix is calculated, all kinds of multimedia information corresponding to the marker could be rendered upon the real world through human-machine interfaces.

2.4 Multimedia information rendering

Currently, our system is implemented to embed the following multimedia information into the real world from the user's view.

(1) Text, such as character description of physical objects and instruction for performing physical tasks in form of annotation.

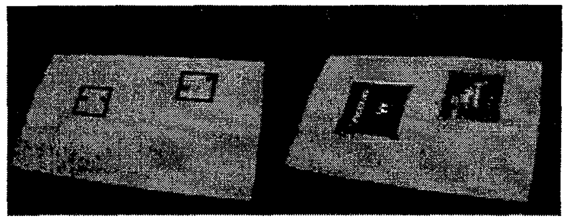
(2) Audio, such as speech instruction

(3) Image.

(4) Real-time video stream.

(5) 3D model, such as OpenGL model and VRML model.

The 12 bits code of the pattern can identify 2^{12} different objects. However, how to decide what computer generated information should be rendered on which object labeled with a square marker becomes an issue to solve. We implemented a database using C++. In the database, the key word for each record is the marker code ID, and the property fields are the media information the system will render when the corresponding marker labeled object runs into the camera's view (Fig. 3).



(a) (b)

Fig. 3. (a) Image of a white board, captured from a CCD camera. (b) Computer generated graphic image superimposed on the white board.

3. Comparing with other visual marking technologies

Some systems, like the Augmented Reality for Construction (ARC) [4], use a 1D barcode as the marker to handle a large number of real world objects. The drawback is that an extra barcode reader is needed for each user. Another similar visual marking method adds color into the 1D barcode. Each bar in a special color represents a certain value. With the same amount of bars, the color barcode can identify more objects than a traditional black and white color barcode. One constraint is that the marker must be located in a daylight or fluorescent environment, otherwise, the color of the bar will not be recognized correctly.

Two existing vision-based AR systems use 2D square marker. In one system [3], the pattern of the marker region is compared with templates, which were given to the system before identifying a specific marker ID. The pattern template could be user names, characters or an image. A significant disadvantage of the method is that it does not have a specific way to differentiate one pattern from the other. The ability to correctly pick up the template matching the marker pattern mostly depends on the resolution of the camera and the algorithm to match the pattern. In another system, Jun Rekimoto proposed a method to use 2-D matrix code marker as landmark in his vision-based system [5]. The complexity of detecting the 2-D barcode and extracting bar code ID depends on how the 2-D matrix code marker is defined. Compared to the marker definition method we proposed, the 2D matrix barcode has some limitations. (1) Every marker has a guide bar aside, which helps to locate the four corners of the marker and determine the orientation of the marker. So, an extra step is needed to find the guide bar. Since the marker code is a binary code, error correction is possible to implement. The problem is that once an error bit is detected, the system will have to skip the current image and extract the marker code from the next image instead of correcting the error bit. In our experimental system, we found that the error bit doesn't occur independently from one image to another in that in most cases the error is caused by the camera capability, lighting in the environment or the material made of the marker. Therefore, dropping the currently processed image when an error occurs will probably not get the correct marker code in the consequent video images. A good method to solve the problem mentioned above is to add a block sum checking function into the pattern recognition algorithm. Block sum checking can correct any single 2bit error in the 2-D binary code block.

This raises a question. Why do we use a square block for checking? In a vision-based AR application, the object labeled with a marker could be of any size. In order to make our system recognize not only big but also small objects, we should minimize the marker size as far as possible. Suppose the marker code is a N-bit code with X rows and Y columns. The number of parity bits is X+Y. The total code is X x Y + X + Y bits. In order to minimize the total code length, X must be equal to Y.

$$X \times Y = N$$

$$X \times Y + X + Y = N + X + N / X$$

$$(N + X + N / X)' = 1 - N / X^2$$

$$1 - N / X^2 = 0$$

$$X = Y = \sqrt{N}$$

Next, we will discuss performance issues of our real-time vision-based AR prototype using the enriched 2-D matrix code marker. Based on the new Binary Square Marker recognition algorithm, the system can recognize multiple objects in real-time (12 frames/sec) on a Pentium III workstation over a MS NT4 platform. We also run the prototype on the Xybernaut MA IV wearable computer with a 233MHz Pentium MMX, which does not support OpenGL graphics. The processing rate there is limited to 3 frames/sec. Actually, more than half of the processing time is consumed in graphic rendering. This observation implies that augmenting a computer graphic hardware onto the wearable computer can improve the performance significantly.

4. Potential applications

In this section, we propose several potential applications using AR and marker recognition techniques. Those applications could be used in industry, entertainment and education.

4.1 Tracking

The Binary Square Maker could be used as 3D mouse in a Virtual Environment (VE), such as the CAVE™ system. The user in the VE holds an object with a marker tagged on it. A camera is put in a location where the marker will be in the camera's view when it moves in a certain area. When the user moves around, the camera can track the position and orientation of the marker, which means that the 6 DOFs of the user's hand are extracted from the known geometry of the Binary Square Marker pattern. More than one mouse could be tracked simultaneously, if the marker codes of each mouse are unique.

4.2 Collaborative architecture design in a shared space

In doing industrial training, attending a conference or playing a game, people often work together face to face. That is a kind of traditional work style and most people intend to keep this tradition because by working in a shared space, such as a classroom or workshop, one can interact with others directly which will obviously improve the work efficiency. Augmented reality provides computer-generated information to the real world in a 3-D space. It helps users perform collaborative real world tasks. One potential application is Collaborative Architecture Design. Some markers representing virtual objects, such as an office building, a parking lot, a residence or a gym, are put on a big table. Several users wearing wearable computers and HMDs look at the big table. Users see graphic images of the virtual objects superimposed on the associated markers through the HMDs. The users can discuss the design of the campus and relocate any facilities either by moving the marker associated with the facility to a new location, or by translating or rotating the virtual objects through user interfaces such as keyboard and mouse. Those pose changes of one user will be distributed to the other users through a wireless connected server. The other users will be able to see the virtual objects in a changed position.

4.3 Mobile tour guide system

By combining augmented reality technology and the mobility of wearable computers, AR applications could be used anywhere beside the research laboratory. Wearing a wearable computer which has an AR system installed, a person can do many computer aided real world tasks in a work room, museum, or even outdoors. In this part, we will introduce a mobile tour guide system, which could be used in an art museum. It is difficult for a visitor to understand the art pieces in an art museum very well without a guide. Although the visitors wish they could have their own guide so that they can navigate in the museum freely, this is not very practical. Even if the guide could talk to the visitor face to face about the work of art, the information the guide provides is very limited. The mobile tour guide system compensates the limitations of human guides. When the visitor sees the marker labeled on the art piece, for example a painting, s/he will hear the music of the author's time from the earphone, see the photo or video of the author through the HMD, etc.

5. Conclusions

In this paper, we addressed the problem of identifying a great number of real world objects with a robust marking approach by using computer vision AR techniques. In the first part, we gave an introduction to the augmented reality technology. Then, we described some existing problems in this research area. In the second part, we presented the workflow of our prototype using a new marking approach. After that, a comparison among our solution and other visual marking technologies was made. The binary code error detection and correction functions used in the marker recognition algorithm make the lighting-sensitive, vision-based application more robust. In the rest part of this paper, we introduced some potential augmented applications in industry, education and entertainment, all of which are based on the vision-based augmented reality environment we built.

We are currently building a more scalable collaborative augmented reality application. The whole system will be running on new hardware with stronger computer graphic capabilities.

References

[1] R.T. Azuma, *SIGGRAPH '95 Course Notes: A Survey of Augmented Reality.* Los Angeles, Association for Computing Machinery, 1995.

[2] Y.Hung, P.Yeh, and D.Harwood, *Passive Ranging to Known Planar Point Sets,* Proceeding of IEEE International Conference on Robotics and Automation, Vol.1, St.Louis, Missouri, 25-28 March, pp.80-85,1985.

[3] M. Billinghurst and H. Kato, *Collaborative Mixed Reality.* In Proceedings of International Symposium on Mixed Reality(ISMR '99). Mixed Reality--Merging Real and Virtual Worlds, pp. 261-284.

[4] A. Webster, S. Feiner, B. MacIntyre, M. Massie, and T. Krueger, *Augmented Reality in Architectural Construction, Inspection, and Renovation.* In Proc. ASCE Computing in C.E., 1996.

[5] J. Rekimoto, *Matrix: A Realtime Object Identification and Registration Method for Augmented Reality.* In Proc. of Asia Pacific Computer Human Interaction (APCHI '98)

[6] J. Rekimoto, and Y. Ayatsuka, *CyberCode: Designing Augmented Reality Environments with Visual Tags, Designing Augmented Reality Environments*. In DARE(2000)

[7] W.N. Hoff, A. *Computer vision based registration techniques for augmented reality.* In Proceedings of Intelligent Robots and Computer Vision XV, SPIE Vol.2904, Nov 18-22, 1996, Boston, MA, pp. 538-548.

[8] D. Koller, G. Klinker, E. Rose, D. Breen, R. Whitaker, and M. Tuceryan, *Real-time Vision-Based Camera Tracking for Augmented Reality Applications.* In Proceedings of the Symposium on Virtual Reality Software and Technology (VRST-97), Lausanne, Switzerland, September 15-17, 1997, pp. 87-94.

[9] I. Herman, *The Use of Projective Geometry in Computer Graphics.* Lecture Notes in Computer Sciences No. 564, eds. G. Goos and J. Hartmanis, Springer Verlag (1992).

[10] O. Faugeras, *Three-Dimensional Computer Vision A Geometric Viewpoint.* Cambridge, MA: MIT Press, 1993.

[11] N.I. Durlach, *Psychophysical Considerations in the Design of Human-Machine Interfaces for Teleoperator and Virtual-Environment Systems.* In Medicine Meets Virtual Reality II: Interactive Technology & Healthcare: Visionary Applications for Simulation Visualization Robotics (pp. 45-47). San Diego, CA, USA: Aligned Management Associates.

[12] E.Sharlin, P.Figueroa, M.Green and B.Watson. *A Wireless, Inexpensive Optical Tracker for the CAVE™.* In Proceedings of IEEE Virtual Reality 2000 Conference, 18-22 March 2000,New Brunswick, NJ, pp. 271-278.

[13] Y. Cho, J. Lee, and U. Neumann, *A Multi-ring Color Fiducial System and An Intensity-invariant Detection Method for Scalable Fiducial-Tracking Augmented Reality.* In IWAR, 1998.

[14] A. State, G. Hirota, D.T. Chen, B. Garrett, and M. Livingston, *Superior Augmented Reality Registration by Integrating Landmark Tracking and Magnetic Tracking,* SIGGRAPH 1996, New Orleans, LA, pp.429-438, 1996.