Physically Based Animation In An Augmented Reality Environment

Guenter Schwann
schwann@icg.tu-graz.ac.at

Computer Graphics and Vision
Graz University of Technology
Graz / Austria

Abstract

This paper gives an overview of Augmented Reality. In the first part it shows what Augmented Reality is, and why to use it. Then various techniques needed and commonly used in AR systems are described. In the second part one particular setup is described with it's hard and software. And with two little applications it is shown what major problems exist in typical AR situations and what solutions may help in the future.

KEYWORDS: Augmented Reality.

1 Introduction

1.1 Definition

What is Augmented Reality? In general it is every technique which extents the reality with some virtual effects. It can be thought of as somewhere between completely real and Virtual Environment[8]. Fig. 1-1 shows the relationships of real, augmented and virtual. But this includes many things. For example this definition includes the computer generated visual effects in movies. In common the AR is seen like in Ronald Azuma's "A Survey of Augmented Reality"[3]. AR systems have to follow three characteristics:

1. Combines real and virtual
2. Interactive in real time
3. Registered in 3-D

Most implementations of AR use some sort of tracker (see section 2) with 6 degrees of freedom and a HMD (head mounted display). But this definition does not insist on an HMD.

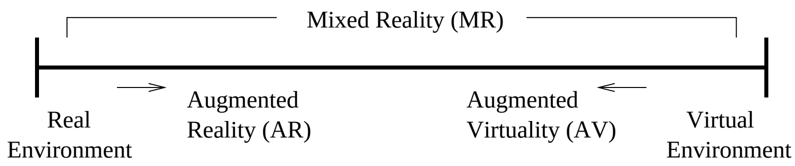

Figure 1: Simplified representation of an Reality-Virtuality Continuum

1.2 Motivation

Augmented Reality is very interesting because combining real and virtual worlds enhances user's perception. Enhancing the perception can also enhance its interaction with the real world. This helps to perform real-world task faster and better. Some examples are shown in the next section.

Research on AR is a challenge. There are many things to improve and to find out. Accurate tracking, the display and the user-interface are the greatest problems at the moment. The EMMIE project[4] of the Columbia university for example addresses the user-interface-problem. EMMIE is a hybrid user interface combining a variety of different technologies and techniques, including virtual elements, and physical objects such as tracked displays and input devices. The prototype includes additional 2D and 3D displays, ranging from palm-size to wall-size, allowing the most appropriate one to be used for any task.

1.3 Examples

Annotation Information that is helpful is added to the real world. One example is Steven Feiner's prototype that helps navigating through the campus of the Columbia university[6]. In museums additional information to the objects would be very useful.

Entertainment Games and interactive TV are only two areas where AR could be used. And it's much cheaper to make virtual objects than physical ones. AR makes it possible to create completely new types of games.

Manufacturing and Repair HMDs could help the technicians if they have to handle very complex machines. This could save a lot of time. Boeing for example is working at this area.

Medical The doctor can look at the real patient and additional data can be shown. Or the doctor can see inside the patient before the operation. And two doctors on different locations could work on one virtual body.

Military The soldiers can walk through unknown terrain, and the correct path is shown in a HMD. Or data as height and speed are shown the pilot of an aircraft.

Robot path planning The user manipulates a virtual robot whose actions are added to the real world, and thus can be verified.

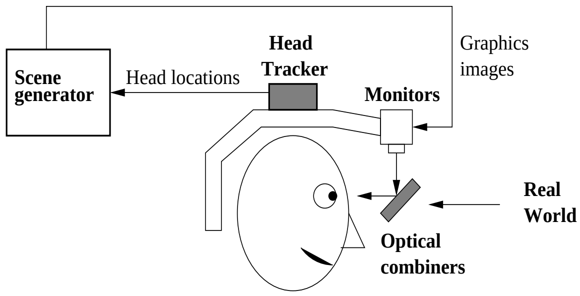

Figure 2: Optical see through HMD

2 State of the Art

Augmented Reality needs three basic subsystems. Those are the same as in Virtual Reality.

- Scene generator
- Display device
- Tracking and sensing

Rendering is not a major problem of AR because in most application only a few objects have to be drawn. The display device and tracking are very sensitive parts of the system. If the tracking is imprecise or the a low-quality display is used, the illusion of augmention is destroyed. So various techniques to handle these two problems are described in this section.

2.1 Display Device

2.1.1 See through HMD

See through HMDs use optical combiners to mix the real world's image, and the virtual image from monitors. Fig. 2.1 shows a simple diagram of a see through HMD. The opaque displays reduce the amount of light from the real world by about 30%.

See through has the advantage, that the resolution of the real world is not limited by the resolution of the displays. In most applications there are only a few virtual objects. It isn't that bad, if only the virtual objects are displayed in low resolution. So the resolution of the displays is not a major problem.

There is a delay between the real and the virtual image. The image of the real world is seen without any delay. But the virtual image has a lag, because it must pass the tracker, the scene generator and the monitor. This end-to-end system delay depends of the used hardware. A 50 ms delay is typical to midrange systems used today. This delay gets only a problem if there is some motion. So this is a dynamic error. One method to reduce this effect, is to predict the users next position[2].

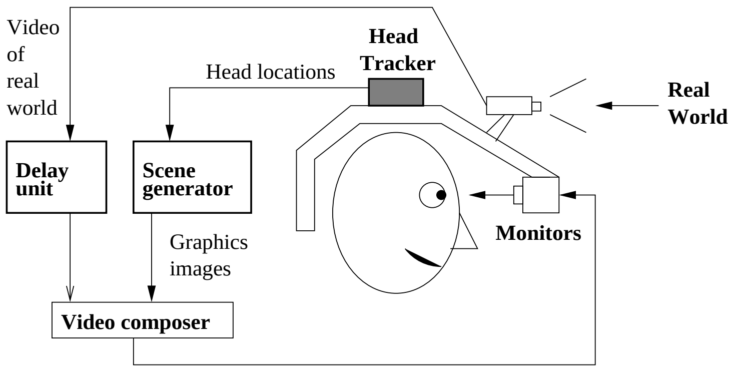

Figure 3: Video see through HMD

2.1.2 Video see through HMD

Video see through HMDs use a closed view HMD. Those closed view HMDs are well known from virtual reality. Two cameras are mounted on the head, and the virtual image from the scene generator is combined with the image delivered by the cameras. Fig. 2.2 shows the concept of the video see through HMDs.

There are no dynamic errors with this device, because the real image can be delayed to have the same lag as the virtual one. But the delay between mechanical motion and the seen motion can cause motion sickness. Closed loop Augmented Reality (tracking and augmention is performed on the same image) can be done rather easily, because no additional camera is needed. A big disadvantage is that the real world image has the same (low) resolution than the display has. But it's easier to control visual behavior like brightness or shadows because the real world's image can be manipulated. And the virtual image can complete overpaint the real image. With through HMDs only bright objects can overpaint only the reality, because 30% of the real worlds image and 70% of the virtual image can bee seen in the display. In bright environments even rather bright objects cannot overpaint the real objects completely.

2.1.3 Monitor based

Monitor based systems are also possible. It's similar to the video see through HMD. But the Monitor is fixed, and the cameras and the tracker are mounted on a robot[3]. The user can control the robot. But this type of system is not very common. Projection walls or Cave systems can also be used for Augmented Reality. The ARGOS[5] project of the University of Toronto uses a monitor based AR system to control a robot.

2.2 Tracking

Mechanical Mechanical trackers have a very high accuracy and reliability. But they are not flexible, and have a very limited range and are limited to one user. Fakespace produces some mechanical trackers.

Magnetic Magnetic trackers use one fixed transmitter, and sensors. They are used rather often, because they are very robust. But the error increases with larger distances. They are also disturbed by ferro-magnetic objects and other magnetic fields. The tracking systems of Polhemus are magnetic trackers. Another magnetic tracking system is to use the magnetic field of the earth. This system shows very bad accuracy, but is not limited in it's range and very easy to setup.

Optical Optical tracker aren't that robust but have a very high accuracy. And they need a powerful computer to keep tracking time short. There are two ways of tracking. Mounting a camera on a helmet and tracking fixed features (for example LEDs on the wall). Or mounting features on the helmet tracked by fixed cameras.

Acoustic Acoustic tracker systems use ultra sonic signals. But to work properly, a large number of transponders is needed on known location. The system delay is rather high. As the optical tracker the acoustic needs line of sight to work. Possible systems are the Intersense IS-600 / IS-900.

GPS The Global Positioning System (GPS) can be used for outdoor tracking. It should work on every place on earth. But it's sensors need contact to the sattelites. This is a problem in buildings. It's accuracy is very limited to about 10-100 meters. Differential GPS can achieve accuracy of about 0.1-1 meters. But this is by far not enough for most applications.

Inertial Accelerometers and gyroscopes are used to record users movements. Because the position can not be recorded directly but is integrated over all recorded data. The result tends to drift with time and gets imprecise. This sort of system is well suited for outdoor applications in combination with portable systems.

Hybrid Hybrid tracker try to combine the strengths of at least two other tracking methods. All combinations are possible. Examples are inertial and optical or magnetic and optical tracking like in Auer's work[1]. Hybrid systems are complex because two systems have to be handled and combined. But they show the best results, and will be the widely used in the future.

3 Development Environment

3.1 Software

The Software base is "Studierstube". It's a project started on the Vienna University of technology, which tries to build an environment for collaboration in Augmented Reality[11]. It's very flexible. It runs on SGI Irix, Windows NT and is ported to Linux. At the moment three different tracker-types are supported: Ascension Flock of Birds, Polhemus Fasttrack, and Intersense IS-600 / IS-900. As output-devices see through HMD (Virtual IO i-glasses, Sony Glastron) and Virtual Table (a workbench-like device) are possible. For 3D rendering Studierstube is using SGI OpenInventor. To enable multiuser applications a system called distributed OpenInventor[9] is running on top of Inventor.

Studierstube also tries to find a high quality user-interface[10]. It's using Personal Interaction Panel (PIP). PIP is a two-handed interface to control Studierstube[9]. It consists of a small panel (about 20x20 cm) and a pen which can be used as "3D mouse". Both of them are separately tracked. But the application can "replace" those tools with anything that is needed, by overpainting it in the HMD-display. Most times the pen remains as a "3D mouse". And on the panel some 3D GUI-widget like sliders and buttons are placed to control Studierstube.

For tracking Studierstube uses a trackerserver that multicasts the position and orientation of each sensor to all connected hosts. This is not the fastest method, but it's very flexible. The hosts can be separated, and two persons can work on one project at the same time, and can be in different rooms or even cities.

The applications are only shared libraries loaded when needed. This has some advantages. For example it's possible to run several applications simultaneously[7]. And it's possible to have several instances of the same application running. To make this possible, Studierstube has to implement some techniques that are known from operating systems.

If more than one application is running, than there are some equivalents to windows in 2D for 3D needed. Studierstube has also build in something like 3D-windowing - a box-shaped volume. Unlike it's 2D counterpart, one 3D-window can be shared between users to allow collaboration.

3.2 Hardware

The tracking-system used is ascension's Flock of Birds. It has 4 sensors. One on the panel, one for the pen, and two for the two HMDs. Each sensor is attached to an electronic unit. The Transmitter is attached to one of these boxes, and one of the boxes is connected via RS232 to a 486-33MHz PC running Linux. The PC multicasts the 6 degrees of freedom from each sensor to the two rendering machines. The rendering machines are SGI O2 workstation. Each workstation delivers the images for one HMD. The HMDs used are IO Display Systems I-glasses. Figure 3-1 shows the complete setup.

The panel is a simple 20x15 cm wood-tray. The pen is a 20 cm long plastic-pipe. It has one button connected to the PC's parallel board. The exact manufacture is not that important. They both should only be light-weight and comfortable to "wear".

Flock of Bird uses DC magnetic tracking which is not that badly disturbed by ferro-magnetic objects as AC magnetic tracking is[1]. According to the specifications the tracker measures up to 72 inch (182 cm). It's static position accuracy is 0.02 inch (0.5 mm) with a standard deviation of 0.07 inch (1.8mm). It's Orientation accuracy is 0.1 with a standard deviation of 0.5. Both at a distance of 12 inch (30,5 cm).

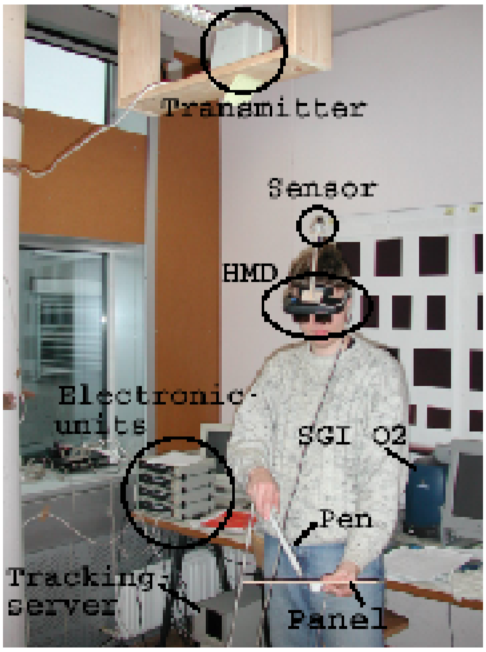

Figure 4: Hardware setup

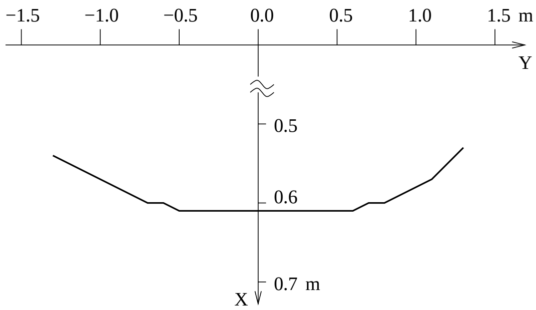

Figure 5: Distortion of the magnetic field

The IO Display Systems I-glasses work in 3D mode. For 3D each eye needs an extra image. Those two images are transmitted line-interleaved. The display's transparent LCD have 180.000 pixels for 225x266 lines of resolution. The field of view is 30 degrees for each eye.

4 Experiments

4.1 Magnetic-tracker Test

It is important to know it's equipments limits. So testing the magnetic tracker is good to see the accuracy of this tool in the environment it will be used.

A post mounted on a plate was used. In a height of 1.58 meters a sensor was mounted. The Flock of Birds transmitter is mounted in 2.19 m height. This measurement unit was moved along the Y-axis from -1.3 m to 1.3 m. The steps the

Figure 6: R2D2 disappearing behind the door

unit was moved is 10 cm. At every step 5 measurements where done and the mean value taken to avoid wrong results influenced by jitter. The result can be seen in Figure 4.1. It can be easily seen, that the tracker works with high accuracy while the distance to the transmitter is not more than 0.5 - 0.6 meters. If distance to the transmitter is more than about 1.0 m the distortion is significant and getting worse and worse with further increasing distance. The results show, that the field gets more disturbed on the + side than on the - side. One possible reason is the near distance to the wall and a column with lots of iron inside.

One effect not shown is the jitter becoming worse with increasing distance. Sometimes values coming from the tracker are completely wrong. This is not easy to measure, but can be seen best, when watching a 3D model showing the position and orientation of a sensor. The model is suddenly jumping to another place and is back in the next moment.

4.2 Starwars

A small application was taken to implement a program with pure Studierstube (that means without windowing, distribution, and other complex techniques). The application simulates a light-saber known by the movie Starwars, and a robot (let's call him R2D2) should walk on the corridor outside. R2D2 should be seen only if he is in the door.

The saber is a simple model consisting two cylinders. One for the sword, one for the hold. Geometry and material is set with OpenInventor. The position of the saber is taken from a Tracker-engine. Studierstube provides the OpenInventor-engine[12]. The engine holds position and orientation of every sensor. The position and orientation of the sensor mounted on the Pen is taken. So the Pen feels like the hold, and the user sees the light-saber instead.

The program itself is very simple. There are only some lines of code, that start Studierstube and load one OpenInventor file. The rest of the work is done in the file loaded. There the saber is made, and connected to the trackerengines output that corresponds to the pen. The robot is made, and a shuttle transformnode lets R2D2 walk. The interaction of R2D2 and the door is made with two cubes that

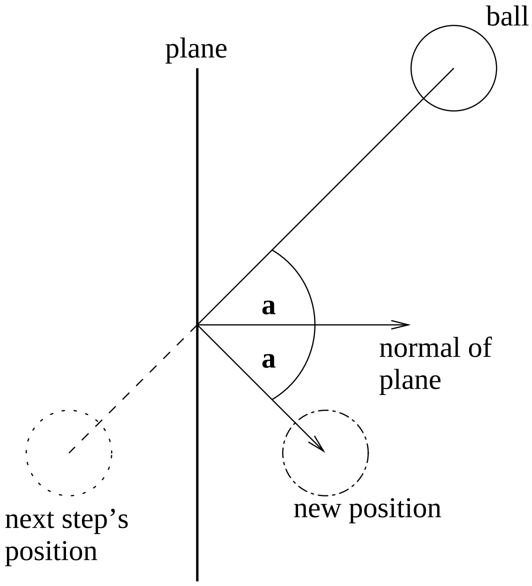

Figure 7: The ball's reflexion at a wall

are at the position of the walls. The cubes are drawn black and therefore can't be seen in the display. R2D2 is automaticly drawn correct by the rendering engine.

4.3 Jumping Ball

A second application is jumping ball. A ball with 5cm radius starts falling from 2 m down to the floor. Gravity is taken into account, and if the ball touches the floor or any furniture the ball rebounces. The ball also has some speed for x and z direction.

The program has an object "ball". This object holds the direction, position, gravity, take off power and routines for calculating the next position, the effect of gravity and reflection at a plane. A general routine calculates if the ball will touch any furniture, the floor, or any wall. An exact model of the office room is needed with all its furniture, monitors and computer to make this possible. The model is stored in an OpenInventor file as an indexed face-set.

To calculate if there is a reflection the ball is reduced to a point. This not correct, but is sufficiently accurate for this demonstration application. The point of intersection of the line defined by the balls position and direction with every plane in the model is calculated. If the intersection point is nearer than the distance the ball will make in the next step, then an reflection could happen. To check this it must be seen if the point is inside the triangle or outside. Therefore for all three triangle edges it is checked if the point is inside or outside.

If a reflection occurred then once again it is checked if another reflection happens. But in the second control, the point of intersection, and the rest of the distance is taken. The second control must be done, because otherwise it could happen that the ball leaves the room in the edges.

Gravity ,take off power and the time-slots used for calculating the balls next position can be changed easily. It would be an option to change these values by sliders on the panel. With this method the user could change the speed and the balls behavior. Even a racket controlled by the pen could be implemented to make a virtual office squash.

5 Conclusion and future work

The test with the magnetic field indicates, that the useful range of the magnetic field is limited to about 0.5 m. By simply using the application Starwars, this can be verified. If the position and orientation isn't accurate enough anymore , the robot disappears before the door ends, or walks into the wall. Displacement of the saber can also be seen easily. But that's not only the static error of the tracking system. The system delay can be seen when moving the pen left and right in a rather high frequency. The frequency can be chosen such, that if the pen goes right, the saber goes left, and the same the other way round.

Another problem even noticeable in Starwars is the limited field of view. When holding the saber upright, the user can't see both ends of it. The head has to be moved. But many users try to look down, left or right instead of moving the head. Building a mask is a possibly a solution. Because the user can't see the bright environment, and is forced to look through the darker glasses.

Following the jumping ball is a hard task because of the limited field of view, and the system latency. It seems that at least low cost Augmented Reality setups are not well suited for applications with fast movements at the moment.

Possible solutions for the problem with the high system latency could be a hardware upgrade. This is already partly done for the tracking server. And it shows that the dynamic error has decreased dramatically. A prediction unit[2] anticipating user's movements would bring further improvements. The other components as network interfaces and rendering engine should work fast enough for such small applications.

The static error could be reduced with correction curves. Many points with correction data would be needed. Those datas must be tracked very precise, probably mechanically tracked. To correct a volume of $2 \times 2 \times 1$ meters with a resolution of 10 cm and 10, 4851 points would be needed. Each point needs $36^3 + 3 = 46.659$ measurements. This would result in more than 10^8 measurements.

One further problem is detection of the real world. The Starwars application for example does not recognize when the door is shut, and R2D2 is still drawn. Optical systems could detect changes in the environment, but this task seems to be to complex at the moment.

The future work on AR in general, will focus on develop simpler hybrid tracking systems, combining AR with wearable computers and research on powerful GUI-systems.

References

[1] Thomas Auer. *Hybrid Tracking of Augmented Reality.* PhD thesis, technical university of Graz, 2000.

[2] Ronald Azuma and Gary Bishop. Improving static and dynamic registration in an optical see-through hmd. In *Proceedings of SIGGRAPH '94.*

[3] Ronald T. Azuma. A survey of augmented reality. In *Presence: Teleoperator and Virtual Environments 6.*

[4] Andreas Butz, Tobias Hllerer, Steven Feiner, Blair MacIntyre, and Clifford Beshers. Enveloping users and computers in a collaborative 3d augmented reality. In *Proceedings of IWAR '99.*

[5] David Drascic, Julius J. Grodski, Paul Milgram, Ken Ruffo, Peter Wong, and Shumin Zhai. Argos: A display system for augmenting reality: Abstract. In *ACM SIGGRAPH 1993.*

[6] Steven Feiner, Blair MacIntyre, Tobias Hller, and Anthony Webster. A touring machine: Prototyping 3d mobile augmented reality system for exploring the urban environmen. In *Int. Symb. on Wearable Computing 1997.*

[7] Anton Fuhrmann and Dieter Schmalstieg. Multi-context augmented reality. Technical report, Institute of Computer Graphics, Vienna University of Technology, 1999.

[8] Paul Milgram, Haruo Takemura, Akira Utsumi, and Fumio Kishino. Augmented reality: A class of displays on the reality-virtuality continuum. *SPIE*, 1994.

[9] Dieter Schmalstieg, Anton Fuhrmann, Ger Hesina, Zsolt Szalavari, L. Miguel Encarnacao, Michael Gervautz, and Werner Purgathofer. The studierstube augmented reality project. Technical report, Vienna University of Technology, 2000.

[10] Dieter Schmalstieg, Anton Fuhrmann, and Gerd Hesina. Bridging multiple user interface dimensions with augmented reality. Technical report, Vienna University of Technology, 2000.

[11] Dieter Schmalstieg, Anton Fuhrmann, Zsolt Szalavari, and Michael Gerautz. Studierstube - an environment for collaboration in augmentetd reality. Technical report, Vienna University of Technology, 1996.

[12] Josie Wernecke. *Open Inventor Mentor.* Addison Wesley, 1994.