A Virtual Reality Browser For Space Station Models

Michael Goldsby, Abhilash Pandya, Ann Aldridge
Lockheed Engineering and Sciences Company
2400 NASA Rd. 1, mail code C-44
Houston, Texas 77058
goldsby@graf10.jsc.nasa.gov

James Maida
NASA, Lyndon B. Johnson Space Center
maida@graf6.jsc.nasa.gov

ABSTRACT

The Graphics Analysis Facility at NASA/JSC has created a visualization and learning tool by merging its database of detailed geometric models with a virtual reality system. The system allows an interactive walk-through of models of the Space Station and other structures, providing detailed realistic stereo images. The user can activate audio messages describing the function and connectivity of selected components within his field of view. This paper presents the issues and trade-offs involved in the implementation of the VR system and discusses its suitability for its intended purposes.

1. INTRODUCTION

This paper describes a Virtual Reality browser made at the Graphics Analysis Facility (GRAF) at NASA/JSC in Houston.

The GRAF is a service and research group providing a variety of graphics services to NASA and NASA contractors. In broadest terms, the GRAF provides the means for its customers to visualize the products of their imagination. It helps them visualize equipment that does not yet exist and maneuvers that have yet to be carried out. The GRAF maintains an extensive repertoire of detailed geometric models of the Space Shuttle, its payloads, and the various Space Station designs in different stages of construction. From them, it produces color printouts, transparencies and animations on video tape.

A VR capability finds a place as an extension of the work the GRAF already does. It gives the customer another means of viewing the models. Its advantages and drawbacks compared to the other media used at the GRAF are discussed below.

2. DESCRIPTION OF THE SYSTEM

2.1. Hardware

The system's hardware components are shown in Figure 1. They consist of two Silicon Graphics Reality Engines, up to two more Silicon Graphics workstations, an Apple Macintosh Super Mac personal computer, a Virtual Research head-mounted display (HMD) with earphones, and two Bird magnetic trackers. The Reality Engines are used to render a stereo pair of images which are routed to the displays in the HMD. The Mac plays pre-recorded messages on request, which are heard through the earphones of the HMD.

2.2. Operation

Wearing the HMD, the user can "fly" through one of the geometric models in the GRAF database (see Figure 2). A magnetic tracker positioned on top of the HMD gives his current direction of gaze. Another tracker is used as a "throttle" to control the user's travel. In the present system, the user may only travel in the direction of gaze with a forward or backward sense. The throttle tracker is held in the hand and has only three meaningful positions: forward (palm down), backward (palm up) and stationary (palm sideways). (An earlier version of the system allowed the throttle tracker to set direction and speed of travel, but most of us VR neophytes found that too confusing.) The user can inquire about the object in the center of his field of view. If there is a pre recorded message corresponding to that object, it is played through the headphones of the HMD. The message may include information about the function and connectivity of the selected object. The user presently registers the inquiry through the mouse of one of the Reality Engines, but work is in progress to replace the mouse button clicks with throttle-hand gestures. A click on the right mouse button arms the inquiry, causing a large red cross to appear at the center of the field of view. Another click of the right button selects the object lying under the cross, or a left button click disarms the inquiry; in either case the cross disappears.

2.3. Software

2.3.1. Software Structure

The system's software is structured as a set of servers and a single client. The client may be thought of as representing the "ego" of the user. There are two drawing servers (one for each eye), a tracking server, and a sound server. In addition, a small sound production program runs on the Mac. A diagram of the software components is given in Figure 3. Each drawing server produces one member of the stereo image pair. The tracking server reads the Bird trackers and returns the current orientation and position of the user. The sound server conveys requests for pre-recorded message to the Macintosh.

An advantage of using a client-server approach is that the client and servers may be distributed about the local-area network for performance or convenience. Each drawing server runs on one of the two SGI Reality Engines. The tracking server runs on another SGI workstation because the trackers happen to be connected to it. The client and sound server can run on any workstation on the network.

2.3.2. Conversion of an Existing Drawing Program to VR Use

An in-house program called DMC is used by the GRAF to view the models on the workstation screen and to produce hardcopy stills and animations. This same program is used to draw the images for the VR system. It might be of interest to anyone considering doing something similar to know what we had to go through to convert an existing drawing program to VR use.

DMC already produces 3D perspective projections from user-selected viewpoints. To make it usable for VR, it was necessary to do the following:

1) drive DMC from the client program rather than interactively from the workstation keyboard;

2) make the projection match the geometry of the HMD; and

3) make the viewing transformation match the current user position and orientation; and

4) synchronize the appearance of the new images for the two eyes.

It proved easy to make these changes, in part because we had access to the DMC source code.

3

The first item can be accomplished in various ways, for instance by creating an additional UNIX process to take the place of the user. By making some simple additions to the source code we could do it even more directly, by making the entire VR episode an extended user command.

To make the projection match the HMD geometry, we followed the outline of Robinett and Rolland [1]. The Virtual Research helmet uses the LEEP optics, the same as the VPL EyePhones described in [1]. In both systems, the optical axes do not pass through the center of the LCD display screens. DMC did not allow off-center viewing windows; that capability had to be added to it. We made no correction for the optical distortion introduced by the LEEP system [1].

DMC already computes the viewing transformation according to user-specified position and orientation.

On double-buffered workstations, image synchronization can be obtained by synchronizing the swapping of the buffers. To separate the swapping of the graphics buffers from the rest of the drawing procedure required changes to the DMC source code. Because we sometimes had to run the two drawing servers on machines of different speed, synchronization was clearly necessary. (It was probably advisable in any case, since clipping can cause a difference in the complexity of the two images; the off-center viewing windows only tend to increase the difference.)

2.3.3. Data Structures for the Selectable Messages

In order to draw a correspondence between the pre-recorded messages and the visible images, it was necessary to superimpose another level of organization on the model data. In this case it was easy to do, because the describable units corresponded to subtrees of the tree representing the image. We simply kept a list of describable elements (read in from a file, hence easily altered) in which the node name of the graphical subtree for the element was listed opposite the file name of the pre-recorded message. For more involved interactions with the model, however, it seems likely that describing the connection between the interaction and the displayed image could well be the most difficult part of the problem.

3. TRADE-OFFS

The models built by the GRAF are extremely detailed. For that reason, it takes a long time to draw them. Refresh times of a second on the SGI Reality Engine are not uncommon. Because of the low resolution of the displays in the HMD (208 by 139 pixels), most of that detail is lost, anyway. Because of the great redraw time, the illusion of continuity is lost. The slow refresh and low resolution interfere with the sensation of immersion. However, the latest models (and they change daily) with their detailed geometry are what our customers are interested in seeing.

The media typically used by the GRAF now are printed images, transparencies and animation recorded on video tape. Our VR system may be compared to these media:

	real-time selection of viewpoint by the user	continuity	resolution	stereo images	sound
VR	Y	N	low	Y	Y
animation	N	Y	high	N	Y
stills	N	N	high	N	N

The VR system has the advantage of providing immediate access to a large collection of models of interest to the GRAF's customers. Also, it provides features that other media lack (stereo images and real-time viewpoint selection). Moreover, it was easy to implement, since it uses an already-existing drawing program. But it has the disadvantages of low refresh rate and low resolution.

4

4. FUTURE DEVELOPMENT

Several avenues of future development are planned. One is to represent the user by a human model in the environment. The GRAF has a long-standing involvement in biomechanical research and modeling and maintains an anthropometrically correct human modeling capability. The model will be used to do reach studies and is also expected to increase the sense of immersion.

Movement within the virtual environment will also be made more convenient. An "elevator" mechanism will be added to allow the user to rise and fall without changing his head orientation, and a reorientation mechanism will be implemented to allow the user to change his head-up direction. Some more intuitive way of flying will be devised, perhaps using a joystick or a third magnetic tracker. At present, the system reads the orientation of the head-tracking sensor but ignores its position; position movements will be included so that the user can look around an object without having to fly around it.

We are also investigating ways to increase the refresh rate of the system (see Appendix).

We plan to study the effectiveness of the system as a tool for visualization, learning and design. Several of the designers of the Single Launch Space Station have shown interest in the system and said that they found it helpful to view their designs in it. Their comments made apparent the need for better motion control. They also expressed a desire for lower lag time (no surprise to us) and higher resolution.

ACKNOWLEDGMENTS

REFERENCES

[1] Warren Robinett and Jannick P. Rolland, "A Computational Model for the Stereoscopic Optics of a Head-Mounted Display", *Presence*, Vol. 1, No. 1, Winter 1992.

APPENDIX

Several approaches to speeding up the drawing rate suggest themselves:

> 1) Rebuild the models to simplify them.
> 2) Change the drawing algorithm or the model data structures.
> 3) Algorithmically simplify the models.
> 4) Switch among smaller local models depending on the user's position.

Rebuilding the models would be uneconomic; it would require more effort than it is practical for us to expend on it.

The drawing algorithm and data structures are already efficient; changing them without changing the number of polygons to be drawn would not change the drawing speed very much.

It is reasonable to ask if there is any kind of algorithmic simplification which could be applied to the data, either off-line or on the fly. (One sort of algorithmic simplification is common and is used in this system, namely clipping. It is viewpoint-dependent, hence must be done on the fly.) For instance, might it be possible to change the level of detail of the model or the resolution of the image depending on the viewer's position? Note that simplifications such as these might imply changing the data structures or drawing methods.

An algorithmic simplification at a gross level would be to switch among different local models as the user moves about. For instance, when the user is outside the station, all data structures for the interior detail might be omitted, and when he is inside a module, all structures exterior to that module might be omitted.

Figure 2 - Virtual Reality in Use.

8

Figure 3- Virtual Reality Software

9