[Published in Human-Computer Interaction, Proceedings of Interact-90, August, 1990.]

INTERACTIVE SCIENTIFIC VISUALIZATION: AN ASSESSMENT OF A VIRTUAL REALITY SYSTEM

Philip J. MERCURIO

Advanced Scientific Visualization Laboratory, San Diego Supercomputer Center

Box 85608, San Diego, California 92138-5608 USA

Thomas D. ERICKSON

Human Interface Group/ATG, Apple Computer, Inc.

20525 Mariani Avenue, MS 76-3H, Cupertino, California 95014 USA

A virtual reality system, consisting of a head-mounted stereoscopic display and a computer-interfaced glove, was assessed by examining interaction with a 3-D model of the human brain. Interactions were recorded on videotape. Non-trivial user interface issues were identified, ranging from constraints imposed by the nature of the wearable interface hardware, to the choice of gestures for controlling the interaction, to problems with a metaphor used in the interface. Some possible solutions are discussed. Effective solutions to these problems, coupled with increases in the computational power of the underlying hardware, are needed for virtual reality to realize its immense potential for scientific visualization.

1. INTRODUCTION

Scientific visualization is a domain of computer science whose goal is to promote visual conceptualization in the process of scientific investigation. As a National Science Foundation report on Visualization in Scientific Computing states: *The ability of scientists to visualize complex computations and simulations is absolutely essential to insure the integrity of analyses, to provoke insights and to communicate those insights with others* (McCormick, et al. (1987)).

Interactivity is fundamental to scientific visualization. The scientific method is by nature an interactive and iterative process, and thus requires interactive graphics environments which interfere as little as possible with the scientist's explorations. Researchers from diverse fields including fluid dynamics (Helman and Hesselink (1989)), medical research (Fuchs, et al. (1989)), and the earth sciences (Hibbard and Santek (1989)) have remarked upon the importance of the ability to interact with their data.

Traditional means of interacting with data, particularly with 3-D data models, leave much to be desired. A common approach to interacting with a 3-D model employs a knob box, an array of (typically 8) continuously-rotatable knobs assigned to parameters of the rendering transformation (typically X, Y, and Z translation and rotation, and scaling). Knob boxes are a cumbersome and non-intuitive means of interaction. Many researchers have investigated alternative means via both software and hardware innovations, with varying degrees of success. Software approaches have included rotation techniques not subject to the pitfalls of X, Y, Z rotation (Shoemake (1985)), techniques for specifying arbitrary origins of rotation (Bier (1986)), and investigations into the use of 2-D input devices for specifying 3-D rotation (Chen (1988)). Hardware approaches have included custom controllers with knobs mounted on orthogonal axes (Mosher, et al. (1986)) and numerous unique interaction devices, such as those created by Fred Brooks' laboratory at UNC Chapel Hill (Brooks (1988)). However, even the most successful of these methods requires going into a special mode to rotate objects, or lacks sufficient correlation to interaction with real-world objects.

Our goal in this investigation was to explore virtual reality as a user interface for scientific visualization. Virtual reality is a computer-generated illusion of a data space in which the user has a virtual presence. One way in which virtual reality has been implemented is with a head-mounted display and a computer-interfaced glove. The user wears a headset containing a binocular video display; the headset also contains a position sensing device, so that as the head moves, the display is appropriately updated, creating the illusion that the wearer is looking around in a static visual environment. The user also wears a computer-interfaced glove which allows hand position and configuration to be incorporated in the interaction with the virtual world. The use of the glove allows the user to directly manipulate objects in the display using natural gestures. It was because of this potential for direct interaction via natural gestures that we felt that virtual reality offered immense potential as a scientific visualization environment.

Our approach was to bring up a scientific data set under a commercial virtual reality system so that we could evaluate it. Although we had already received demonstrations of virtual reality systems, the data sets were created by the vendors, and consisted of relatively simple displays (typically virtual rooms in which a user could move and interact). We felt a more telling evaluation would be to use an existing data set, a 3-D contour model of a human brain.

2. BACKGROUND

2.1. The Virtual Reality Platform

We were fortunate to obtain the cooperation of VPL Research, a leading vendor of virtual reality systems. VPL provided us with access to an experimental version of a single user virtual reality system—it was experimental in that it was currently under development, and also in that the rendering workstations used were of lower power than in the final, commercial system. They also gave us information about file formats and system limitations so that the data set could be appropriately transformed before our visit.

The computer hardware for the virtual reality system consists of a Macintosh II, which processes the positional data received from the head-mounted display and the computer interfaced glove, and two Silicon Graphics Personal Irises, which render the 3-dimensional image of the data set. The interface hardware consists of a binocular video display—the EyePhone , and a computer interfaced glove—the DataGlove.

The “EyePhone” consists of 2 color LCD screens fitted into a rubber mask worn on the face like a scuba mask. Lenses installed in front of the LCD screens magnify the images and help the wearer merge the images into one binocular view. The EyePhone’s rubber face mask blocks all ambient light, so all that the user sees are the images presented on the EyePhone display. The field of view is 100 degrees horizontally, with 60 degrees of overlap, and 60 degrees vertically. The resolution of the EyePhone’s LCD displays is 360 x 240 pixels. Considering the proximity of the LCD’s to the eyes, each pixel subtends a rather large portion of the view, and the resulting resolution is relatively low.

The EyePhone weighs 4.25 lbs, with the weight of the display being counter-balanced by weights positioned behind the head. On the straps joining the display to the weights is a mount for a Polhemus 6-D tracking device. The position in 3-space and 3 degree-of-freedom orientation of the tracker are detected by a nearby rigidly-mounted receiver and transmitted to the computer managing the virtual reality. The EyePhone is connected to its interface box by a long cable.

The DataGlove is a glove fitted with fiber optic sensors to measure the amount of bend in the joints of the fingers, and its position is tracked via a separate Polhemus tracker. Like the EyePhone, it is connected to its interface box via a long cable.

2.2. The Virtual Reality Interface

In the virtual reality environment that we studied, there are two distinct ways of moving the model being viewed. One is moving about under your own locomotion. That is, if the model is floating in front of you, you can walk toward it and through it. If you are above it, you can bend over or sit down. From a stationary position, you can move your head to alter the direction of viewing. The other means of movement is gesture-controlled “flying.” By pointing with your forefinger extended and the rest of your fingers curled in you fly forward, in the direction you are looking (not the direction you are pointing). Pointing with two fingers causes you to fly backwards.

The virtual reality system provides feedback on the position and configuration of the user’s hand, as sensed by the DataGlove. When the gloved hand is within the user’s field of vision (or what would be the user’s field of vision if the user could actually see), the display shows an image of a hand in the position and configuration sensed by the DataGlove.

A third means of movement is also possible, but was not available in the version of the system which we examined. If you reach out with your hand and the image of your hand in the virtual reality intersects an object, you can make a fist and grab the object, and move it by moving your hand.

2.3. The Data

The brain data was originally acquired by embedding a human brain in paraffin. Successive thin layers were shaven off of the block of paraffin and the top of the block was photographed onto movie film in a process called “cinemorphology”. The resulting film frames were then hand-digitized into a computer to produce 3-dimensional stacks of 2-D contours. The contours for different neuroanatomical structures were digitized separately and can thus be displayed independently or in selected groups. The resulting brain database consists of about 75,000 vectors, or, when converted to polygonal surface data, 150,000 polygons (Livingston and Wilson (1976)).

2.4. Data Preparation

The display limitations of the graphics hardware in the system described above is approximately 1200 polygons (at the rates necessary to achieve satisfactory interaction). To cull the brain database down to a manageable size, the contour data for each section of each brain structure was first replaced by the orthogonal rectangle bounding that section, reducing the number of vectors representing that section to four. This produced a 3-D stack of aligned 2-D rectangles which were then converted to a polygonal mesh consisting of four 3-D rectangles for each section.

Even this crude, boxy representation of the data did not make it possible to view the entire set of brain structures simultaneously. The final data set brought into the virtual reality consisted of the brainstem, cerebellum, thalamus, red nucleus, and the top inch (approximately) of the cerebral cortex, with Broca’s and Wernicke’s language areas of the left hemisphere represented as separate objects. This data set consisted of approximately 1000 polygons.

2.5. Evaluation Method

The virtual reality system was qualitatively evaluated by the two authors: a scientific visualization specialist who was familiar with the data set, and a user interface specialist who had had only slight exposure to the data set. Each evaluator used the system for about twenty minutes, with the interactions being recorded on videotape.

3. DISCUSSION

In this section we begin by giving a capsule description of the experience of interacting with the virtual reality environment. We then survey the problems we found Ñ some of which arise from current (although not inherent) hardware limitations, and some which arise from the design of the interface.

3.1. The Interactive Experience

When the headset was put on and the system started up, the user could see a large brain (scaled up by an order of magnitude) floating in space . Although the resolution of the display was poor, the stereopsis and interactivity offered by the virtual reality environment enhanced the reality of the brain image to the extent that we quickly lost awareness of the low resolution. We were, however, continually aware of the boxiness of the image, which resulted from the transformation of the data set from 150,000 polygons to the roughly 1,000 that the rendering engines could handle in real time.

Movement in the virtual reality via real-world locomotion and head motion worked quite well. The correspondence between real-world movement and movement in the virtual world was so accurate that there was no need for a conscious translation of intent to the user interface action necessary to achieve that intent. It was very natural to walk towards the brain image, have it get bigger, and then be inside it. There was virtually no learning necessary to interact with the data set in this way. The only difficulty with this type of movement was that, in spite of the greatly simplified data set, there was still a bit of jerkiness when the image moved in sync with the user's head movements.

Flying was more problematic. First, due to a miscalculation, the speed of movement was set too high. Since this version of the virtual reality system didn't give the user interactive control over speed, flying to a particular spot was very much a hit or miss proposition. Usually, several forward and backward passes were required to get close to the target. Since the brain image was only about ten feet in diameter, flying inside it was not necessary; however, since the brain contours were opaque, this made flying much more difficult, since the target could not be seen until it was present. While flying was the most difficult part of the interaction, it still took only about two minutes to get the basics down.

3.2. Hardware and System Limitations

First of all, there are a number of pragmatic factors that spring from the fact that the user is wearing the interface hardware. The user wears the EyePhone upon his or her head. Although it is reasonably comfortable despite its weight of 4.25 lbs., the user is aware that tilting the head too far to one side could cause the unit to slip off. Because the EyePhone blocks all ambient light, and because the virtual reality display does not give any feedback about the external environment, the user must remain aware of a number of physical constraints. The cables connecting the EyePhone and DataGlove to their interfaces boxes are long, and can become tangled or twisted around the wearer. The user must also be aware of the location of the stationary Polhemus receiving units and must not walk too far from them or too close to them. Walls may also pose a problem. In this instance observers were present who were able to warn us about physical obstacles and constraints so that we were able to operate with little attention to these factors. However, for virtual reality to be a truly practical form of interaction, it will have to be usable without supervision. For example, the virtual reality environment could also reflect salient features of the surrounding physical environment.

Another limitation was the inability of the rendering hardware to handle more than about 1200 polygons without serious degradation in performance. Because of this, the brain data set had to be greatly simplified. The original data beautifully represents the folds and contours of the brain structures. All of this detail was lost in exchange for the ability to display more of the brain structures. The structures that were chosen were those which best survived the translation to boxy solids. It should be noted that this limitation is due only to the capabilities of the available graphics hardware; with sufficiently powerful real-time rendering machines, much more detailed data can be displayed.

One of the major interaction problems evident in the videotape is with the speed of movement through the data set. If the user flew, the user flew very, very fast. This particular problem was due to the fact that the software that was in development at the time we used it, and that the speed of movement through the data was linked to the scaling of the data.

Another hardware limitation is the slow update rate and unsteadiness due to the temporal and spatial resolution of the Polhemus tracking devices. This problem is not inherent in the virtual reality environment itself, and will be alleviated by advances in tracking hardware.

3.3. User Interface Problems

While the virtual reality system makes use of gestures to fly through the virtual environment, the gestures used were problematic. One problem was that the gesture used to fly (pointing with one finger to fly forward) is a relatively natural one. Both of us flew inadvertently a number of times, either because we were trying to point at something as we spoke about it, or because other habitual gestures (e.g. finger on the chin; stroking the mustache) would sometimes map onto the "fly" gesture. A second problem is that, although the gesture is a natural one, it has no meaningful connection with the command it invokes. Such semiotic gestures are difficult to learn and remember, although this is not a problem in a system with only a few gesture commands. A better example of the use of gesture is placing a "hand" inside an "object", and closing the hand to "grab" it. Examples of some possible solutions are given below.

The problem with the speed of flying has already been discussed. A related problem is that you fly in the direction in which you are looking, rather than in the direction you're pointing. Flying in the direction you're looking prevents users from scanning the image as they move. That is, users should be able to look around while flying. For example, with a very detailed brain image, users might want to move slowly along the fissure between the temporal and frontal lobes, watching for the speech area. But that would be off to the side, perpendicular to the direction of movement; since the resolution is not very high, and the angle of view not that wide, it may be difficult to see it with peripheral vision.

A third problem is inherent in the verb "fly," which VPL uses to describe the mode of moving through the data without the user moving in real space. As Lakoff and Johnson (1980) have noted, a single word can evoke a whole series of expectations and beliefs, a metaphoric system. Unfortunately, flying is a poor metaphoric system for movement through a virtual reality environment. There are two reasons.

First, while "flying" is a sexy concept, the fact is that it doesn't feel like you are flying toward the object when you point at it. Rather, it feels as though the object is coming towards you. Presumably this is because there is natural kinesthetic feedback associated with movement, but no such feedback in the virtual reality. While it can be argued that users can suspend their disbelief and ignore their kinesthetic feedback, the fact is that in a very short time they'll want to be paying attention to such feedback as they walk around and through the object, or grab it to reposition it. An interface which requires users to rely heavily on kinesthetic feedback one moment, and ignore it the next, seems problematic.

The second problem with flying as a metaphor stems from the role of metaphor in an interface: the purpose of an interface metaphor is to allow users to apply some of their real-world experience to areas of the new domain with which they might otherwise have difficulty (Erickson, 1990). In this context, the problem with the flying metaphor is that we don't know how to fly in the real world. As a result, there are no intuitive gestures for flying and it is not clear what should control the direction of flight.

3.4. A Possible Solution

It is important to emphasize that the user interface problems—as with the hardware limitations—are not inherent in the virtual reality approach. As evidence of this, we offer one possible solution to a number of the above problems.

Many of the problems noted above may be solved by choosing a different metaphor for moving through the data set. Instead of a metaphor in which the user is depicted as moving, a metaphor in which the user stayed stationary and the data set moved would be more suitable. For example, a user could push and pull the object rather than flying through it. A "push-pulling" metaphor avoids both problems of the flying metaphor: there is no contradictory kinesthetic feedback, since it is the world that is moving; and users know how to push and pull things in the real world.

Unlike flying, push-pulling also suggests some intuitive gestures for doing it: palm open, fingers together, pushing in the direction you want the object (or space) to move; clenched fist, moving in the direction you want to pull the object or data space. As with real pushing and pulling, the action should only continue as long as the hand is moving. The use of such mimetic gestures, combined with a push-pulling metaphor, also eliminates any ambiguity about what should determine the direction of movement: clearly the push-pull should act in the direction of the gesture, not in the direction of view.

Another indication that the push-pulling metaphor is a good one is that it is extensible. For example, momentum could be introduced, so that if you grab something and open your hand before stopping (i.e. throwing), the object would coast by until you grabbed it again. This would also solve the speed problem, by providing an intuitive means for the user to control speed—if you want to go fast, you throw fast. Another possibility that comes to mind is that if a system were configured with two data gloves, two-handed pushing and pulling could translate into shrinking and stretching, thus providing a way to scale the image.

Finally, the natural gestures suggested by the push-pulling metaphor make much more use of the unique features of the DataGlove. In the virtual reality system we used (since grabbing wasn't working), there seemed to be little reason to use the glove—a little box with two buttons for fly forward and backward would have worked as well.

The solution presented here is not ideal. It glosses over a number of problems. For example, suppose the data set consists of a number of separate objects: if a user grabs an object and pulls, does it move the object relative to the other objects, or does it move the entire set of objects relative to the user. The answer isn't obvious. The point here is not to offer a final solution, but to demonstrate that virtual reality environments have non-trivial interface problems which must be, and can be, addressed.

4. CONCLUSION

To read popular descriptions of virtual reality systems, one would think that there are no interface issues. Virtual reality is just like real reality, except better: a wave of the hand, a simple, natural gesture, and whatever the user had in mind happens. Such is not the case. As we have seen, there are non-trivial interface problems associated with virtual reality environments: wearable interface devices may encumber the user; inadvertent gestures may cause unintended actions; gestures may be non-intuitive; metaphors may be inappropriate. While we have suggested some solutions for these problems, the solutions are intended to indicate the nature of the design task, rather than to be the final word.

Displaying the brain data in the virtual reality was both disappointing and exciting. It was disappointing because the transformations to the data left the final image so coarse that it was little more than a curiosity. It is difficult to imagine doing useful scientific research with the data in this form. But while the hardware platforms underlying virtual reality environments are not yet sufficiently powerful to support scientific visualization of the sort of data set we have explored, they are powerful enough to offer an interactive testbed for research on the interface issues.

Nonetheless, the experience was exciting because there is clearly immense potential for displaying data in this manner. We were able to figure out how to navigate through the data with about fifteen seconds of instruction, and a few minutes of trial and error (which was necessary to learn how to cope with the time lag and circumvent some of the problems described above). Even with its problems, virtual reality was by far the easiest and most useful way in which we've been able to interact with this data. And, most importantly, none of the problems was inherent in either the system or the interaction techniques. With careful redesign of some aspects of the user interface, and with increases in the capabilities of the hardware, virtual reality systems will open new vistas in scientific visualization.

ACKNOWLEDGEMENTS

We would like to acknowledge the cooperation of VPL Research, who provided us information, technical support, and access to a version of one of their virtual reality systems which was still under development. VPL Research may be contacted at 656 Bair Island Road, Suite 304, Redwood City, California 94036.

REFERENCES

Bier, E. A. (1987). *Skitters and Jacks: Interactive 3D Positioning Tools*. In Proceedings of the 1986 Workshop on Interactive 3D Graphics (Chapel Hill, NC, October 23-24, 1986), 183-196. ACM: New York.

Brooks, F. P. (1988). *Grasping Reality Through Illusion¾Interactive Graphics Serving Science*. In Proceedings of CHI '88, 1-11 (Washington, May 15-19). ACM: New York.

Chen, M., Mountford, S. J., and Sellen, A. (1988). *A Study in Interactive 3-D Rotation Using 2-D Control Devices*. Computer Graphics, 22, #4, 121-129.

Erickson, T. D. (1990). *Working with Interface Metaphors*. The Art of Human-Computer Interface Design. (Ed. B. Laurel). Addison-Wesley: in press.

Fuchs, H., Levoy, M. and Pizer, S. M. (1989). *Interactive Visualization of 3D Medical Data*. Computer, 22, #8, 46-52.

Helman, J. and Hesselink, L. (1989). *Representation and Display of Vector Field Topology in Fluid Flow Data Sets*. Computer, 22, #8, 27-38.

Hibbard, W. and Santek, D. (1989). *Visualizing Large Data Sets in the Earth Sciences*. Computer, 22, #8, 53-59.

Lakoff, G. and Johnson, M. (1980). *Metaphors We Live By*. The University of Chicago Press: Chicago and London.

Livingston, R. B. & Wilson, K. R. (1976*) The Human Brain: A Dynamic View of its Structures and Organization*. Film, 28 minutes. Presented by Roche Laboratories ; made by Wexler Films. Distributor: Wexler Film Productions.

McCormick, B. H., DeFanti, T. A. and Brown, M. D. (1987). *Visualization in Scientific Computing*. Computer Graphics, 21, # 6.

Mosher, C. E. Jr., Sherouse, G. W., Mills, P. H., Novins, K. L., Pizer, S. M., Rosenman, J. G., and Chaney, E. L. (1987). *The Virtual Simulator*. In Proceedings of the 1986 Workshop on Interactive 3D Graphics (Chapel Hill, NC, October 23-24, 1986), 37-42. ACM: New York.

Shoemake, K (1985). *Animating Rotation with Quaternion Curves*. Computer Graphics 19, #3, 245-254.