ICAT 2003
December 3-5, Tokyo, JAPAN

A Voice and Mouse Input Interface for 3D Virtual Environments

David L. Kao Steve Bryson

NASA Advanced Supercomputing Division
NASA Ames Research Center
M/S T27A-2, Moffett Field, CA 94035, USA
{David.L.Kao, Stephen.T.Bryson}@nasa.gov

Abstract

We propose a voice and mouse input interface for 3D interaction in virtual environments. One of our requirements is that we want a user interface that would work naturally with standard equipment. We performed an experiment to measure the efficiency of using voice in addition to the mouse input device for acquiring target objects randomly placed in the 3D virtual environment. Our preliminary study indicates for some users the voice and mouse interface shortens the time to pick an object in three-dimensional space.

Key words: Voice and Mouse Input Interface, Virtual Reality, 3D Displays, Visualization

1. Introduction

There have been many successful stories on how 3D input devices can be fully integrated into an immersive virtual environment. Electromagnetic trackers, optical trackers, gloves, and flying mice are just some of these input devices. Though we can use existing 3D input devices that are commonly used for VR applications, there are several factors that prevent us from choosing these input devices for our applications. One main factor is that most of these tracking devices are not suitable for prolonged use due to human fatigue associated with using them. A second factor is that many of them would occupy additional office space. Another factor is that many of the 3D input devices are expensive due to the unusual hardware that are required. For our VR applications, we want a user interface that would work naturally with standard equipment. In this paper, we propose a voice and mouse interface for use in 3D virtual environments.

2. Related Work

Previous studies have reported that using voice in addition to an existing input device (e.g. keyboard and mouse) improves user interactions significantly for several applications. In [1], studies have shown that using voice and mouse input together can reduce task completion time by as much as 56% with an average reduction of more than 21%. The task measured is the time it takes users to create drawing with a graphical editor. The studies measured voice in addition to the mouse input. Voice input has also been used in conjunction with eye tracking to reduce pilots' cognitive and manual workload. In their studies, voice recognition and eye-tracking were integrated with aviation display systems [2]. There have also been studies where voice input is used for on-board car navigation systems and assistance. Spontaneous speech recognition input was used to enter spontaneous navigation queries that are recognized, parsed and then replied using a map display [3].

3. Voice and Mouse Input Interface

In our multimodal interface, voice input is used to perform coarse cursor movements and menu selections. A 2D mouse is then used to perform the precise cursor movements or object transformation such as rotations, translation and scaling. This model is similar to the two-hand interaction concept, where the non-dominant hand performs coarse placements of the cursor while the dominant hand performs the precise cursor movements. The two hand input concept had been shown to be simpler to use and understand because of the common correspondence to performing tasks in the physical world. In [4], studies have shown that using two hands to control two independent cursors to perform 3D interaction creates more efficient interface.

In the conventional mouse interface, a 2D mouse performs rough cursor placement via mouse acceleration when the mouse is moved rapidly. While this method is very effective in a 2D interface, it is somewhat awkward when mapping 2D motions to 3D. Instead, we use the voice input for rough placement of the cursor. In one example application, we use a spherical coordinate grid for both the coarse and fine cursor movements. For coarse cursor movement, we predefine six locations on the sphere that are the vertices of an octahedron. The positional commands: One, Two, Three, Four, Five and Six would move the cursor to the corresponding vertex

of the octahedron. The precise cursor movement then can be controlled by the 2D mouse. Moving the mouse horizontally would correspond to moving the cursor in the latitude direction while moving the mouse vertically would cause the cursor to run along the longitude direction. The user can also move the cursor inward or outward on the current spherical coordinate grid by pressing the middle mouse button while moving the mouse, which would also change the radius of the sphere and the corresponding octahedron.

4. Experiment

To test the improvement in cursor control provided by augmenting a mouse interface with voice recognition, we developed a target acquisition experiment in the spirit of Card *et. al.* [5]. In this experiment a target is presented to a test subject, and the time required to place the cursor on the target is measured. If this target acquisition time is shorter for the voice and mouse interface than for the mouse alone, we can say that adding voice for coarse cursor placement enhances this task performance.

4.1. Subjects

Five colleagues served as subjects in the experiment. Only two of them have computer graphics background, which is not a requirement for the experiment. All had experience with using mouse daily for their work; however, most of them rarely use a voice input interface.

4.2. Input Interfaces

Two input interfaces were compared. The first interface is a standard mouse device with three buttons. The second interface consists of a voice input device with the standard mouse device. For voice recognition, we used the IBM ViaVoice software. A standard headset with an attached microphone is used by the subjects for voice input.

4.3. Experiment Environment

We used a 3D dome display system called Perspecta, which offers a 360-degree-viewable volumetric 3D display inside a 20 inch glass dome, for our experiment (figure 1). The display system provides a non-intrusive virtual environment where the subjects can perform the experiment tasks using one of the two input interfaces. None of the subjects had ever used a 3D dome display system previously. The test program was developed on a PC running under Windows 2000. The experiment runs at 32 frames per second.

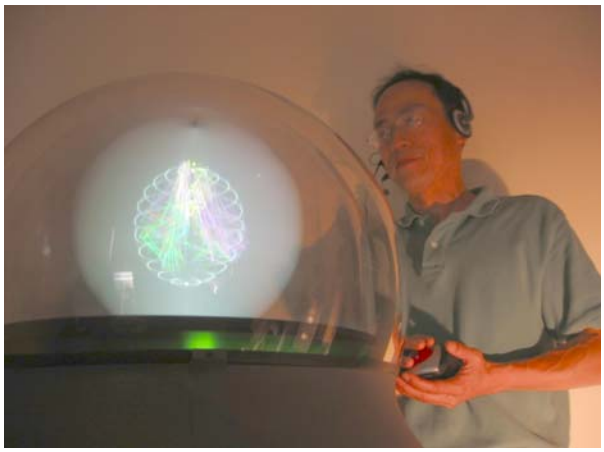

Figure 1. A 3D dome display is used for interactions using the mouse and voice input.

4.4. Procedure

Identical experiments were performed by each subject using both interfaces. Using each interface, subjects were asked to move the cursor to the position of a spherical target randomly positioned inside the 3D dome display. When the test program starts the following objects are shown: (1) the spherical coordinate grid, rendered with latitude and longitude lines, (2) the cursor, rendered as a gray sphere and is always initially positioned at the same location on the spherical coordinate grid at the beginning of each run, and (3) the six landmarks numbered 1 thru 6 at the corresponding octahedron's vertices. The test administrator then initiates the test run. After a random 1/2 to 1 second delay, the test subject is presented with a yellow spherical target object randomly positioned inside the dome display. When the target is presented the timing clock is started. When the cursor and target intersect, to the cursor is rendered as a wire mesh providing user feedback indicating a successful acquisition. To prevent counting "pass-troughs" as target acquisitions, the subject must place the cursor on the target for at least 1 second for a valid target acquisition to be recorded. At this time the clock is stopped and the elapsed time recorded. Then, the test program repeats the above procedure for the next target. After the desired number of repetitions (50 for most of our test) the test administrator ends the experiment. Figure 2 shows a typical scene rendered during the experiments.

4.5 Training

Prior to running the test program, subjects were instructed on the mouse button functions. For voice and mouse input interface, subjects were instructed to speak only the '1' thru '6' voice commands during the test runs. For each run series, the training time for each subject was identified by observing when performance times no longer improved with repetition. The remaining timings are reported below as results.

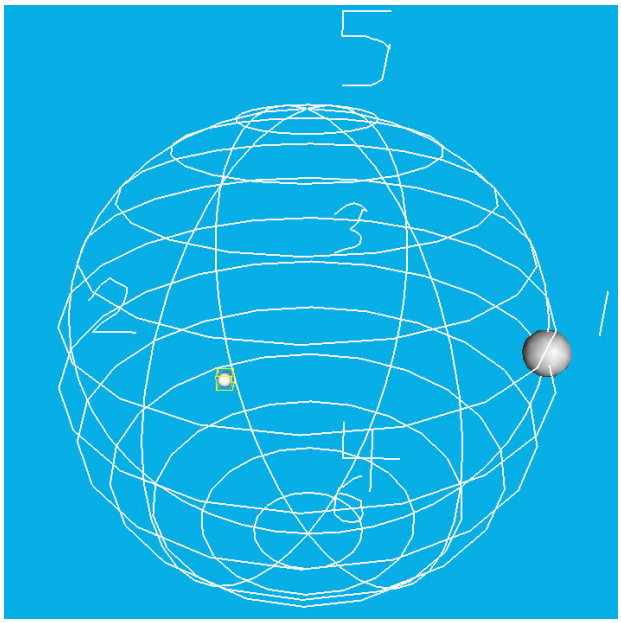

Figure 2. A snapshot of the scene rendered during the experiment. The small yellow shaded sphere is the target and to the right is the probe rendered as a gray sphere. The spherical coordinate grid with latitude and longitude lines are shown with the six landmarks.

4.6 Design Considerations

During the experiment, several test run timings had to be excluded from the results because of unintentional voice input spoken by the test subjects. Furthermore, in order for the IBM ViaVoice software to recognize the individual subject's voice, all subjects were asked to dictate a text passage with 57 sentences prior to the experiments

One important feature that we had implemented in the test program is to avoid the situation where the cursor would get stuck in the window boundaries. This would cause subjects to be frustrated while moving the probe since they are only looking at the 3D dome display instead of the 2D window where the test program is displayed. When this occurs, additional clock time is introduced. We resolved this problem by checking for the condition when the cursor is moved near to the boundary and then resetting the cursor position to the center of the screen using Window's SetCursorPos() function.

Another problem that may occur is when the subject moves the cursor outside of the test program's display window. This may cause undesirable actions (e.g. subjects selected some menu option in the Window desktop.) We resolved this problem by maximizing the test program's window to be full screen size and restricting the cursor to be confirmed within the program's window.

5. Results

Figure 3 gives the average times of task performance for both the mouse only and mouse and voice interface. We can see that while the differences between the two interfaces do not show a strong statistical significance, for subjects 2 and 4 the mouse and voice interface provided a notably shorter task completion time. It is also interesting to note that the difference between the interfaces is smaller for subjects who could perform the task more quickly. This suggests that the usefulness of the voice/mouse interface increases for users who are less skilled in task performance. This result also suggests to us that for more complex tasks the mouse and voice interface would be superior to the mouse only interface, consistent with the results in [1].

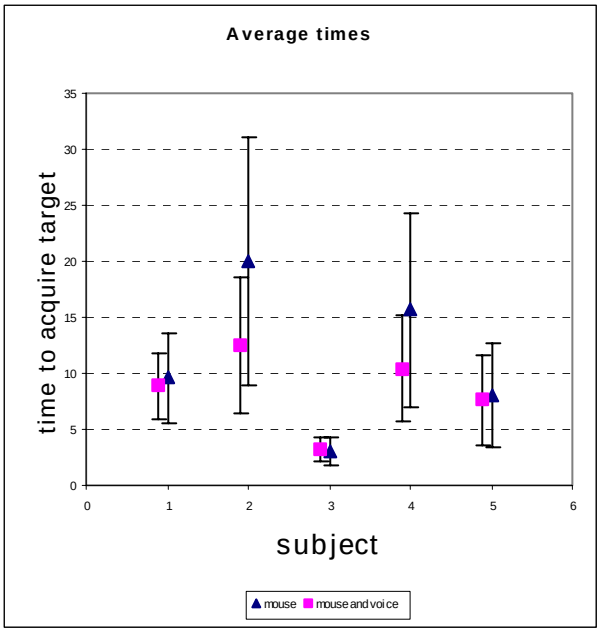

Figure 3. The average times to task completion for the five test subjects. The error bars show plus and minus one standard deviation in the data.

A more sophisticated analysis is provided by plotting the target acquisition times vs. the distance from the initial cursor position to the target. Fitts' law [6][7], a classic result of experimental psychology, finds that the relationship between time to target and distance is given approximately by (for a target of fixed size)

$$time = a + b \log_2(distance)$$

where a and b are constants determined through experiment. Smaller a and b indicate a more effective interface. The correlation coefficient of the line fits of our time vs. distance results were less than that for time vs. $\log_2$(distance) in accordance with Fitts' law. Figures 4a and 4b show the plots of time vs. $\log_2$(distance) for subjects 2 and 4, respectively, which indicate that the voice and mouse interface is notably superior to the

mouse interface.

Figure 5 shows a typical plot for the other subjects where, while there is no statistically significant difference between the interfaces, the voice and mouse interface shows slightly improved performance.

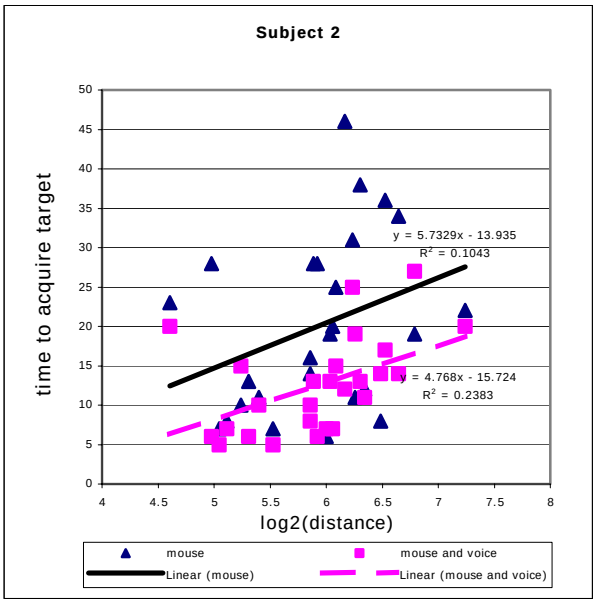

Figure 4a. Plots of time vs. $\log_2$(distance) for subject 2, showing the enhanced task performance of the mouse and voice interface over the mouse interface for the test subject.

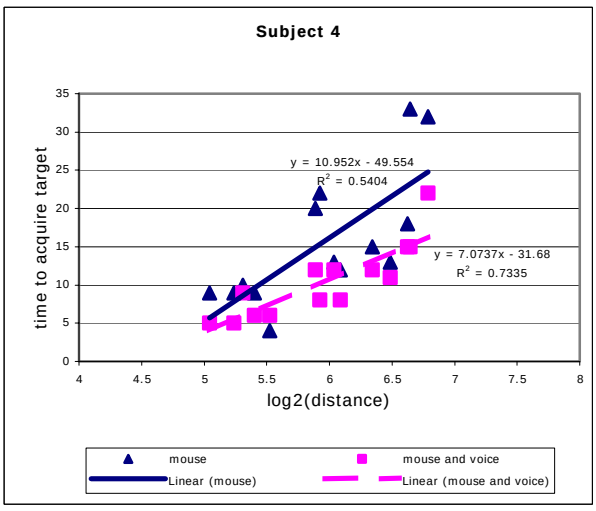

Figure 4b. Plots of time vs. $\log_2$(distance) for subject 4, showing the enhanced task performance of the mouse and voice interface over the mouse interface for the test subject.

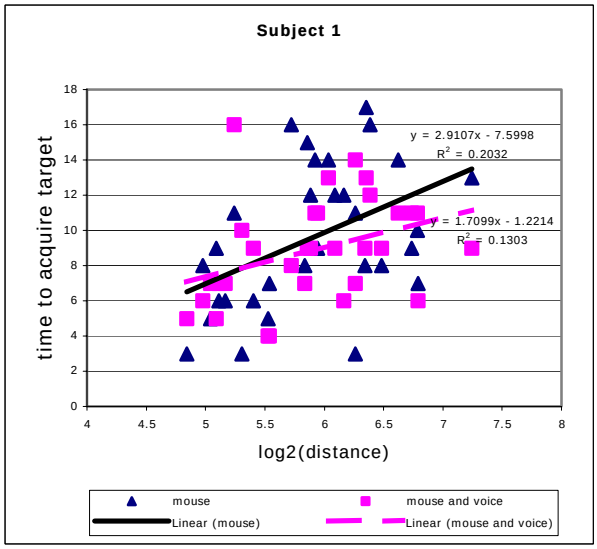

Figure 5. Plots of time vs. $\log_2$(distance) for subject 1, which is typical of the results for subjects 1, 3 and 5. While the difference between the two interfaces is not statistically significant, we can see that the slope of the Fitts' law line fit is slightly lower for the mouse and voice interface.

Table 1 gives the constants *a*, *b* and the correlation coefficient R^2 for the Fitts' law line fit of our results.

	Mouse			Mouse and voice		
Subject	*a*	*b*	R^2	*a*	*b*	R^2
1	-7.6	2.91	0.20	-1.22	1.71	0.13
2	-13.9	5.73	0.10	-15.7	4.77	0.24
3	-1.96	0.88	0.25	-2.47	0.93	0.21
4	-49.6	10.95	0.54	-31.7	7.07	0.73
5	-13.2	3.58	0.24	-5.60	2.23	0.12

Table 1. The parameters of Fitts' law.

6. Lessons learned

From our experiment, we found that there are several improvements that we could implement for future studies. One improvement is to use a better voice recognition software that would allow us to speak short command phrases instead of simple one word voice commands. After the test runs, some subjects complained about the sensitivity of the 2D mouse (e.g. it moves too fast or too slow.) A feature which allows the

user to control the mouse sensitivity prior to the test run would really be helpful. Some subjects also did not like the fact that they can move the probe in both the latitude and longitude directions simultaneously. They thought that this make the control of the probe a slightly more difficult. They would prefer an option to choose so that the probe would move in either one of the latitude or longitude grid lines only.

In our design, the middle mouse is used to changing the radius of the spherical coordinate grid that the probe lies on. Some subjects indicated that they accidentally pressed the left mouse button, which is used for object rotation, instead of the middle button. An improvement would be to use the left mouse for change the spherical grid's radius since the 3D dome display offers 360 degree view and the rotation transformation is used infrequently. Though the spherical coordinate grid used in the experiment seems to be ideal for interaction using a 3D dome display, we can also use other non-spherical grids as well.

7. Applications

We have used the proposed voice and mouse input interface for several scientific visualization applications. In one application, our scientists are able to explore complex graphs of protein interaction networks from the Protein Data Bank using our multimodal interface. These graphs depict protein-protein interactions in yeast. There are several hundred proteins (nodes) and thousands of protein-protein interactions (edges) in the network graphs. Figure 7 shows a typical protein network graph. Each protein is categorized by its functional group. First, the scientist chooses a protein group (via voice input) to view all of the iterations related to this group of proteins. The graphs of these interactions from the selected protein group are displayed with the initial positions of these proteins computed by our graphing algorithm. Some edges may appear to be cluttered due to the number of nodes in the graphs. To reduce the cluttering, the scientist then can use the mouse to control the placement of the edges connected to the selected proteins. Since the network graphs are very complex in nature, the tasks of interactively selecting specific protein groups and manipulating the network graphs would have been very difficult to perform without our multimodal interface.

Though our initial motivation for developing the voice and mouse interface is for 3D interaction applications running in the 3D dome display, our proposed interface can also be appropriate for other 3D virtual environments, particularly when the input task requires high accuracy along some surface. In such a situation freedom of motion in 3D can make tasks difficult to perform, so constraining cursor motion to a 2D surface facilitates task performance. Once motion is constrained to a 2D surface a 2D input device becomes optimal. In another example application, we visualize several biophysical and geophysical data sets measured over the Earth's surface by constructing surface graphs of the measured data over a sphere (the Earth). For each data set, a surface graph is constructed by protruding an amount proportional to the data value radially outward from the sphere. Using our multimodal interface, we can constrain the cursor to move along the Earth's surface so that the data values at the cursor position are shown while the cursor moves. By mapping the current cursor position to the graph surfaces, we can also constrain the cursor to move along the surface graphs. Furthermore, we can easily select one or more surface graphs from the data sets using voice input.

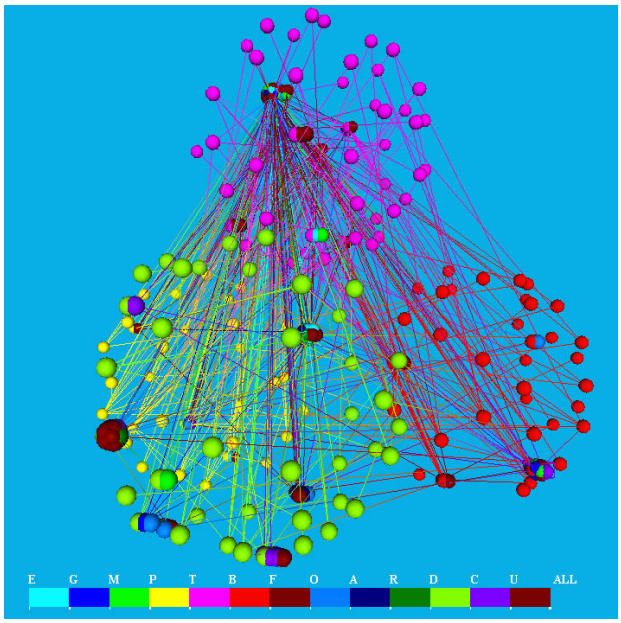

Figure 7. A protein network graph depicting hundreds of interacting proteins. Each edge connecting two proteins indicates that there is an interaction between the two connected proteins. The alphabet labels on the bottom are the function categories of the proteins. The proteins are colored by their function categories.

8. Conclusion

In this paper, we evaluated the effectiveness of adding voice input to the mouse input device for placing the cursor in a 3D virtual environment. Our preliminary study indicated that for some users one can reduce the time it takes to perform the acquiring task when voice is added to the input interface. The results suggest that this effect should become stronger as the task becomes more complex. We found that effective data analysis can be achieved while the scientists view their data rendered inside the dome display and perform user interactions simply using the mouse and voice input.

References

1. R. Pausch and J. H. Leatherby, “An Empirical Study: Adding Voice Input to a Graphical Editor,” Journal of the American Voice Input/Output Society, V9:2, July, pp 55-66 (1991).

2. F. Hatfield, E.A. Jenkins, M.W. Jennings, and G. Calhoun, "Principles and Guidelines for the Design of Eye/Voice Interaction Dialogs," Third Annual Symp. on Human Interaction with Complex Systems, IEEE, pp 10-19 (1996).

3. M. Westphal, A. Waibel, “Towards Spontaneous Speech Recognition For On-Board Car Navigation And Information Systems”, Proceedings of the Eurospeech ’ 99 (1999).

4. R. Zeleznik, A. Forsberg, and P. Strauss, “Two pointer input for 3D interaction”, Proceedings of 1997 ACM Symposium on Interactive 3D Graphics (1997).

5. S. K Card, W. K. English, and B. J. Burr, “Evaluation of mouse, rate-controlled isometric joystick, step keys, and text keys for text selection on a CRT”, Ergonomics, 21, 601-613 (1978)

6. P.M. Fitts, “The information capacity of the human motor system in controlling the amplitude of movement”, Journal of Experimental Psychology, 47, 381-391 (1954)

7. P.M. Fitts and J.R. Peterson, “Information capacity of discrete motor responses”, Journal of Experimental Psychology, 67, 103-112 (1964)