An Evaluation of Techniques for Grabbing and Manipulating Remote Objects in Immersive Virtual Environments

Article · September 1999

DOI: 10.1145/253284.253301 · Source: CiteSeer

2 authors:

Doug A. Bowman
Virginia Tech (Virginia Polytechnic Institute and State University)

Larry F Hodges
Clemson University

An Evaluation of Techniques for Grabbing and Manipulating Remote Objects in Immersive Virtual Environments

Doug A. Bowman and Larry F. Hodges*
Graphics, Visualization, and Usability Center
College of Computing
Georgia Institute of Technology

Abstract

Grabbing and manipulating virtual objects is an important user interaction for immersive virtual environments. We present implementations and discussion of six techniques which allow manipulation of remote objects. A user study of these techniques was performed which revealed their characteristics and deficiencies, and led to the development of a new class of techniques. These hybrid techniques provide distinct advantages in terms of ease of use and efficiency because they consider the tasks of grabbing and manipulation separately.

CR Categories and Subject Descriptors: I.3.7 [Computer Graphics]:Three-Dimensional Graphics and Realism - Virtual Reality; I.3.6 [Computer Graphics]:Methodology and Techniques - Interaction Techniques.

1 INTRODUCTION

A defining feature of virtual reality (VR) is the ability to manipulate virtual objects interactively, rather than simply viewing a passive environment. This capability is desirable in many VR applications and is typically accomplished using a real-world metaphor. Instead of issuing an abstract command or specifying coordinates and rotation angles, users may reach out a hand, grab an object (using a button or a gesture), and move it around the virtual environment (VE) using natural, physical motions [7].

This metaphor has many limitations, however. First, the physical arm is confined to a small space around the user's body. This means that only nearby objects may be grabbed, and the area of object movement is restricted as well. A travel technique [2] is required to allow the user to move to a position where remote objects are close enough to be picked up. Travel may be desirable, as it allows different visual perspectives, but it should not be required.

Also, manipulation of large objects is difficult. They obscure the user's view during positioning tasks, since the user must be within an arm's length of the object to pick it up. As in the real world, the user must place the object, back up to see the "big picture," make corrections, and so on.

* 801 Atlantic Drive
Atlanta, GA 30332-0280
{bowman, hodges}@cc.gatech.edu

In general, the real-world metaphor does not allow efficient, large-scale placement of objects in the VE. In this paper, we discuss techniques that extend the natural metaphor as well as an alternative metaphor. These techniques allow grabbing and manipulation of remote objects (those farther away than an arm's length). Our goal is to identify techniques with which one can reach any object in the environment and completely control its position and orientation in the 3D space (six degrees of freedom). We also wish to use methods that promote ease of learning, ease of use, efficiency, and expressibility.

As we will see, the object positioning task is made up of at least two component interactions: grabbing or selection, and manipulation. Grabbing refers to the initial phase of the task, when an object is picked up. The user must at least specify which object to grab, and may also denote the center of rotation for the manipulation phase. In the manipulation interaction, the user moves the object within the environment, specifying both position and orientation.

2 EXISTING TECHNIQUES

There are a variety of existing techniques which attempt to solve the problem of grabbing and manipulating remote objects. Most of them fit into two categories, which we call arm-extension techniques and ray-casting techniques.

In an arm-extension technique, the user's virtual arm is made to grow to the desired length, so that object manipulation can be done with the hand, as it is with the natural mapping. These techniques differ in the way that the user specifies the virtual arm length. In general, arm-extension techniques make object manipulation simple, because the user moves and rotates the object with natural hand and arm motions. Grabbing is more difficult, because the virtual hand must be positioned within the object, which may be small or distant.

Ray-casting techniques [4] make use of a virtual light ray to grab an object, with the ray's direction specified by the user's hand. The use of the light ray makes the grabbing task easy, because the user is only required to point to the desired object. Manipulation is not hand-centered, however, and is therefore more difficult. In the following sections, we describe techniques of both types that we have implemented and tested.

There are also some techniques which approach the problem in a more indirect manner. Rather than attempting to extend or enhance the natural method, they use various aids to allow manipulation of remote objects. One such technique is the World in Miniature (WIM) [8]. Here, users hold a small representation of the environment in their hand, and manipulate objects by manipulating the iconic versions of those objects in the miniature environment. This can be a powerful and useful metaphor, although the effects of increasing environment size and numbers of objects on the usability of this method are not known.

Another interesting technique scales the user or the entire environment so that any object, no matter its size or distance from the user, may be grabbed and manipulated with the real-world metaphor.

2.1 Go-Go Technique

One arm-extension technique is called the “go-go” technique [6]. Here, a local area is defined around the user at some distance. While the user’s hand stays within that physical distance, the virtual hand moves in a one-to-one correspondence with the physical hand. When the physical hand goes beyond the threshold, however, the virtual hand begins to move outward faster than the physical hand, following a non-linear, increasing function (see Figure 1).

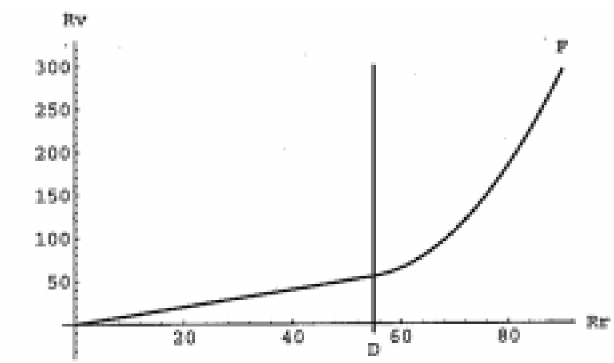

Figure 1. Mapping function for the go-go technique. R_r=physical hand distance, R_V=virtual hand distance. Reproduced from [6].

This technique allows the user to reach objects at a greater distance, while preserving the natural manipulation metaphor. The local area around the body allows fine manipulations to be made in the normal manner. Interaction is completely transparent to the user, since the only actions required are arm motion and grabbing, as in the real-world metaphor. We have also implemented a modified version (“fast” go-go) which has no local area and a more quickly growing function.

The go-go technique, however, still has a finite range, defined by some function of the user’s arm length. For different environments, then, the function used must change in order to allow grabbing of every object in the scene. This may lead to imprecision in the user’s ability to position the virtual arm, so that for quite distant objects, the arm may extend and retract too quickly to allow easy grabbing and manipulation of objects.

2.2 Other Arm-Extension Techniques

We have implemented two techniques which preserve the metaphor of grabbing and manipulating objects with the hand, while allowing infinite stretching of the user’s arm so that any object in any environment may be reached.

First, the “stretch go-go” technique allows infinite stretching using only arm motion for control. The space around the user is divided into three concentric regions, such that the user’s natural hand position is in the middle region. When the user stretches her hand out so that it lies in the outermost region, the arm begins to grow at a constant speed. If the arm is brought back into the innermost region, it retracts at that speed. In the middle region the arm length remains the same. Thus, physical hand position is mapped to virtual hand velocity, with three discrete velocities available. Using this technique, any arm length can be obtained. This is similar to the flying speed control technique discussed in [4]. To aid the user, a gauge is shown which indicates the current region and the proximity of other regions (Figure 2). This technique is more cognitively challenging than the go-go technique, but it does allow the arm to be placed at any distance.

The other technique in this category is an indirect stretching technique. Instead of using arm motions, the user may stretch or retract the virtual arm by pressing buttons on a 3D mouse. One button extends the arm at a constant rate, while another retracts the arm at that speed. This technique takes away the natural metaphor of stretching the arm out to make it longer, but may also be more precise and physically easy to use.

Of course, there are many other possible arm-extension methods. One could map physical hand velocity to the rate of growth of the virtual arm. Velocity control could be added to the indirect stretching technique. The stretch go-go technique could be generalized to allow a continuous range of virtual arm velocities. We have chosen these techniques because of their simplicity and representativeness.

Note that all of the arm-extension techniques give users control over all 6 degrees of freedom for an object. The go-go and fast go-go techniques, however, have a finite distance range, and thus have less power of expression than the arm-extension techniques which allow infinite arm stretching.

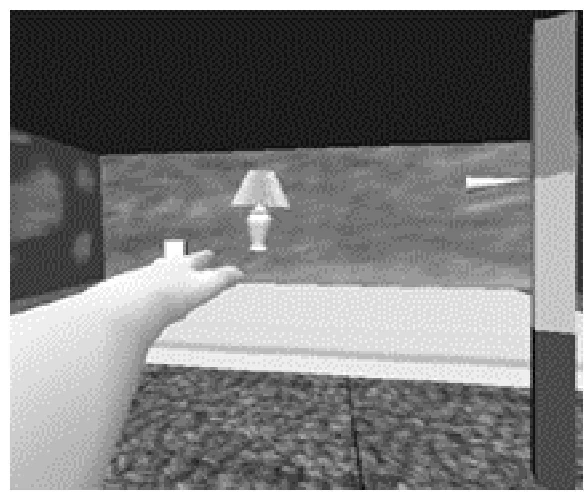

Figure 2. The stretch go-go technique. The gauge on the right shows that he user’s hand is about to enter the outermost region, where the virtual arm will begin to grow.

2.3 Ray-Casting Techniques

As mentioned above, ray-casting is a grabbing technique where a light ray extends from the user’s hand. By intersecting an object with this ray and releasing the button, the object is attached to the ray (Figure 3). Users can then manipulate the object easily using wrist or arm motion.

Ray-casting has several drawbacks, however [1, 6]. First, it suffers from the “lever-arm” problem, meaning that because objects are attached to the end of the ray, there is no simple method for rotating an object in place, except around the axis of the ray itself. Also, it lacks a method for controlling the object’s distance from the user. Thus, only 1 degree of freedom (rotation around the ray axis) may be independently controlled using ray-casting. All other degrees of freedom are dependent on one or more of the others. For example, if the user wishes to move an object vertically (by tilting his wrist upwards), he must also rotate it and translate it in another direction.

In order to allow translation toward and away from the user, we enhanced the ray-casting technique with a "fishing reel" metaphor. After selecting an object via ray-casting, users can reel objects closer or farther away using two mouse buttons, as in the indirect stretch technique described above. Ray-casting with reeling, then, lets users control one additional independent degree of freedom.

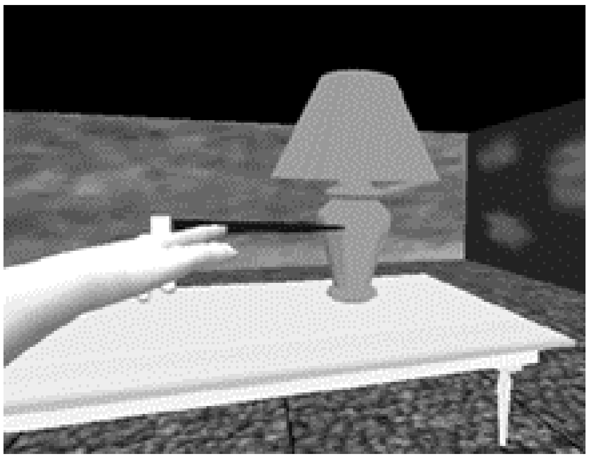

Figure 3. Grabbing an object via ray-casting

3 USER STUDY

An informal usability study was performed using the six techniques described above: go-go, fast go-go, stretch go-go, indirect stretching, ray-casting, and ray-casting with reeling. Users were presented with a simple interior environment containing several pieces of furniture and a virtual human that could all be moved and placed within the room.

3.1 Method

Eleven student volunteers, 2 female and 9 male, participated in the study. A Virtual Research VR4 head-mounted display (HMD) was used, along with Polhemus Fastrak trackers on the head and hand, and a 3-button 3D mouse. The VE was built using the Simple Virtual Environment (SVE) toolkit [3], and was rendered on a Silicon Graphics Crimson with RealityEngine graphics.

Subjects spent as much time as they liked with each of the six techniques, moving and rotating objects within the VE. They were encouraged to make comments out loud, and were asked questions about the relative strengths and weaknesses of each of the techniques.

Although we did not collect quantitative data, the users' comments and difficulties were sufficient to illuminate the properties of the techniques we tested. Further studies with well-defined user tasks and quantitative measures will be performed to support these results.

3.2 Results and discussion

There was no clear favorite among the techniques we tested. The most popular were the go-go technique and the indirect stretching technique, but subjects also noted difficulties with these methods. Table 1 presents a summary of some of our key findings. We have generalized these results into three principles that contribute to the usability of a grabbing and manipulation technique.

First, naturalness is not always a necessary component of an effective technique. Users almost unanimously found go-go to be the most natural technique, but many evaluators preferred other techniques. Indirect stretching was more effective for several subjects because it offered more precise control of the hand location, and less physical work on the part of the user. Several users also liked ray-casting with reeling because of the lack of physical effort required: they could support their arm and simply point with their wrists and press mouse buttons.

Second, physical aspects of users were important in their evaluation of the techniques. For example, those users with shorter arms were less likely to prefer the go-go technique because their reach was more limited. Also, all of the arm-extension techniques depend on the specification of a point at the center of the user's torso. The virtual hand in these techniques is kept on a line defined by this torso point and the location of the physical hand. Although we defined this point relative to the user's head position, the height of the user made a difference. If the torso point is not approximated well, the hand will appear lower or higher than it should be, and grabbing and manipulation will be more difficult. In short, techniques that are dependent on the user will require user modeling in order to be most effective.

Our most important finding, however, was that grabbing and manipulation must be considered separately for overall usability. Although only two of our users preferred a ray-casting technique overall, almost every user commented that it was easier to grab an object using ray-casting than with any of the arm-extension techniques. It requires no arm stretching and less precision on the part of the user: one simply points the ray and releases the button. With the arm-extension techniques, one must place the hand within the object, which can be quite difficult at a great distance or when a small physical motion maps to a large translation of the virtual hand.

On the other hand, no users preferred ray-casting techniques for object manipulation. As discussed earlier, arbitrary rotations of an object are practically impossible using these techniques. With an arm-extension technique, objects can be rotated in their own coordinate system, and their position can be controlled easily as well. None of the current techniques, then, were universally acclaimed, because none of them were easy to use and efficient throughout the entire interaction: grabbing, manipulating, and releasing the object.

Technique	Characteristic	# subjs.
Go-go	finite range	7
Go-go	imprecise grabbing	8
Ray-casting	difficult rotations	11
Ray-casting	can't move objects in/out	11
Ray-Casting	ease of grabbing	10
Arm-Extension	ease of manipulation	9

Table 1. Number of subjects who commented on selected characteristics of the techniques studied

4 HYBRID TECHNIQUES FOR INCREASED USABILITY

Based on user feedback, we have designed and implemented new techniques which are hybrids of the best features from those discussed above, but still retain ease of use and simplicity. These techniques use ray-casting for object grabbing and hand-centered manipulation, since these methods were thought to be the easiest to use and most efficient by almost all subjects. Four users even suggested that such a hybrid technique would be useful.

In the most basic technique, the user grabs the object with the light ray, as before, but instead of the object becoming attached to the light ray, the virtual hand moves to the object position and the object is attached to the hand. When the object is dropped, the hand returns to its natural position. This allows simple grabbing and manipulation with no extra work on the part of the user. We call this the HOMER (**H**and-centered **O**bject **M**anipulation **E**xtending **R**ay-casting) technique.

This combination of techniques is based on work by Mine [5] and Wloka [9], who both use ray-casting in combination with hand-centered manipulation. However, their techniques do not allow specification of an arbitrary object position and orientation, because they use a one-to-S mapping between hand motion and object motion, where S is a scale factor. Therefore, the object may only move in a space S times as large as the extent of the reach of the physical hand.

We offer two extensions to correct this problem. First, object manipulation is done relative to the user's body, instead of absolutely. The grabbed object's position is determined by the vector from the user's body to his hand, and the current distance of the object. Therefore, one can quickly place the object anywhere on a sphere surrounding the user at the current distance, while controlling object rotations independently.

Second, the distance of the object from the user may be specified using either hand motion or mouse buttons. The direct HOMER technique maps the object-to-user distance onto the initial hand-to-user distance, so that positioning the hand twice as far from the body also places the object twice as far away. For more precise control and unbounded distances, the object may be reeled using mouse buttons as previously described, which we call indirect HOMER.

The HOMER techniques offer five advantages over arm-extension techniques. First, object grabbing is easier, since it does not require precise positioning of the virtual arm. Second, objects at any distance may be selected with the same amount of physical effort. Third, object manipulation requires less physical effort. Fourth, objects may be placed at any distance away from the user using the indirect HOMER technique. With direct HOMER, the distance scale factor is implicitly determined by the initial hand-to-user distance, so most distances can be obtained in practice. Finally, object distance is easier to control. Specifying distance using mouse buttons is quite precise, and the linear distance function used in direct HOMER is easier to predict than the non-linear function in the go-go implementation, resulting in more precise prediction of virtual arm length.

5 CONCLUSIONS AND FUTURE WORK

Grabbing and manipulation of virtual objects is a fundamental interaction in immersive virtual environments. Furthermore, it is incorrect to assume that objects of interest will always be within arm's reach, or that users desire to interact with objects solely in their local space. Therefore, it is important that we design techniques for remote grabbing and manipulation, and that we understand their strengths and weaknesses.

In this paper, we have identified proposed techniques such as the go-go technique and ray-casting, and shown their characteristics and limitations through a user study. This study showed that it is important to consider grabbing and manipulation as separate issues. The HOMER interaction techniques take the best aspects of the tested techniques and combine them in a seamless way to maximize ease of use and efficiency.

We plan to continue this research by performing a more formal experiment using the best of these techniques on several specific tasks. These tasks will involve local and remote manipulation, small and large objects and object distances, and arbitrary object rotations. By measuring the speed and accuracy with which users can perform these tasks, we will be able to draw conclusions about the ease of use, efficiency, and expressibility of these techniques.

Finally, all of the techniques considered here operate directly on the objects in the environment. It will also be important to determine in a quantitative and qualitative way the differences between this paradigm and that used in techniques such as WIM or environment scaling, which have unique sets of usability characteristics.

Acknowledgments

The authors would like to thank the volunteers who gave their time for our user study. This work was partially supported by the National Science Foundation.

REFERENCES

[1] D. Bowman and L. Hodges. "User Interface Constraints for Immersive Virtual Environment Applications." Graphics, Visualization, and Usability Center Technical Report GIT-GVU-95-26, 1995.

[2] D. Bowman, D. Koller, and L. Hodges. "Travel in Immersive Virtual Environments: An Evaluation of Viewpoint Motion Control Techniques." to appear in *Proceedings of Virtual Reality Annual International Symposium (VRAIS)*, 1997.

[3] D. Kessler, R. Kooper, and J. Verlinden. "The Simple Virtual Environment Library Version 1.5 User's Guide." Available at http://www.cc.gatech.edu/gvu/virtual/SVE/.

[4] M. Mine. "Virtual Environment Interaction Techniques." University of North Carolina Computer Science Technical Report TR95-018, 1995.

[5] M. Mine. "ISAAC: A Virtual Environment Tool for the Interactive Construction of Virtual Worlds." University of North Carolina Computer Science Technical Report TR95-020, 1995.

[6] I. Poupyrev, M. Billinghurst, S. Weghorst, and T. Ichikawa. "The Go-Go Interaction Technique: Non-linear Mapping for Direct Manipulation in VR." to appear in *Proceedings of the ACM Symposium on User Interface Software and Technology (UIST)*, 1996.

[7] W. Robinett and R. Holloway. "Implementation of Flying, Scaling, and Grabbing in Virtual Worlds." *Proceedings of the 1992 Symposium on Interactive 3D Graphics*, 1992, pp. 197-208.

[8] R. Stoakley, M. Conway, and R. Pausch. "Virtual Reality on a WIM: Interactive Worlds in Miniature." *Proceedings of CHI* , 1995, pp. 265-272.

[9] M. Wloka and E. Greenfield. "The Virtual Tricorder: A Unified Interface for Virtual Reality." *Proceedings of the ACM Symposium on User Interface Software and Technology (UIST)*, 1995, pp. 39-40.