A SPECIFICATION OF 3D MANIPULATION IN VIRTUAL ENVIRONMENTS

S. Augustine Su and Richard Furuta
Hypermedia Research Lab
Department of Computer Science
Texas A&M University
College Station, Texas 77843-3112
E-mail: {su, furuta}@bush.cs.tamu.edu

ABSTRACT

Virtual Reality techniques have promised intuitive and effective user interfaces to virtual worlds. The use of hand gestures is an important part of that interface. However, due to the absence of maturity of standard and tailorable software abstractions such as those seen in 2-D graphical user interfaces, current techniques for specifying the interactions of 3-D objects and gestures are ad hoc and indirect.

In this paper, we discuss the modeling of three basic kinds of 3-D manipulations in the context of a logical hand device and our Virtual Panel Architecture. The logical hand device is a useful software abstraction representing hands in virtual environments. The Virtual Panel Architecture is the 3-D counterpart of the 2-D window systems. Both of the abstractions are intended to form the foundation for adaptable 3-D manipulation.

Within our software framework, the click-and-drag operation from the 2-D graphical user interface context gracefully can be replaced by a meaningful hold-and-move operation for applications in virtual environments. With these tailorable abstraction tools, the semantics of natural and precise gestures can be prototyped rapidly.

INTRODUCTION

Incorporating gestural control into Virtual Reality environments holds the promise of providing intuitive and effective user interfaces to interact with virtual worlds. By using their hands to directly manipulate 3-D objects, the environment's users have the potential to gain much more freedom than in the traditional 2-D mouse and keyboard environments. However, due to the absence of maturity of standard and tailorable software abstractions, current 3-D manipulation techniques are ad hoc and indirect when compared to 2-D graphical user interfaces. Furthermore, since 3-D manipulation is still far from fully explored, the complexity with which current environments permit interactions between the user's hands and 3-D objects is still very limited.

There are two major paradigms for the use of hands in virtual environments. The first paradigm is to point, shoot, or grab 3-D objects. This manipulation method is directly generalized from the use of a 2-D pointer, and can be imple-

mented by a 3-D mouse with buttons, which has the ability to detect positions and orientations in 3-D space. These gestures can be combined with other sources of input; for example, human speech can be combined with gestures to specify quantities as in [1, 2]. In this situation the gestures act as 3-D pointers, and the speech acts as buttons to signify status changes when the hands are not available to push buttons. It is clear that this first paradigm is very useful, but, however, does not take full advantage of the freedom given it in 3-D space.

The second paradigm is to create sets of static or dynamic gesture commands for specific applications as in [3, 4, 5]; each gesture represents a single command with pre-defined semantics in the context of applications. The gestures in this paradigm do not necessarily correspond to physical manipulations—indeed as one example, interfaces can use gestures borrowed from a sign language such as American Sign Language.

Ideal 3-D user interface models have to be able to accommodate not only the above approaches, but also to provide tailorable tools for new user interfaces to meet various needs. We believe we have found a good user interface model for 3-D manipulation. In this paper, we will discuss the modeling of three popular gestures based on a logical hand device and the Virtual Panel Architecture of our work. With proper abstraction tools, the semantics of natural and precise gestures can be prototyped rapidly.

In the next two sections the hand model and the Virtual Panel Architecture will be briefly discussed, respectively. Afterwards, three popular gestures will be described based on the hand model and the architecture.

THE LOGICAL HAND DEVICE

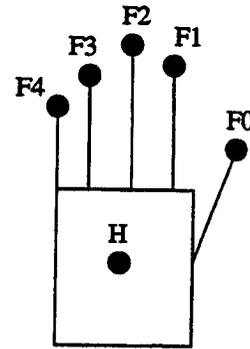The innovation of logical devices in a graphics package is to conceal discrepancies among

Figure 1: The six points of interest on a hand for the hand device

disparate physical devices of a kind, and to furnish device-independent characteristics to application programmers.

By the same token, the logical hand device [6] was designed to be a useful software abstraction representing hands in virtual environments. The hand device reports hand information in the form of events to the system. The hand information consists of

1. the positions and orientations of the five digit tips and the center of the back of the hand (Figure 1); that is, the output of six 3-D mice, or six 3-D pointers.

2. digit-oriented handshape features, such as straight, flat, curved, fully curved, and so on for each finger, and adduction or abduction for adjacent fingers. These features can be used to compose American-Sign-Language-like static gestures.

With this hand device, we can meet the need of the two major paradigms of using 3D gestures in virtual environments: the style of "point, reach, and grab" and the command by sign-language-like gestures.

THE VIRTUAL PANEL ARCHITECTURE

The principle of the manipulation in 2-D graphical user interface is to use a single 2-D pointer

to move into and out of a number of hierarchical 2-D windows, and to use mouse buttons to signify status changes. Based on that, higher-level tasks, such as click-and-drag, can be implemented. This 2-D manipulation methodology can be generalized for 3-D manipulation. Think about the use of hands or fingertips to directly manipulate 3-D objects while the hands are characterized by the logical hand device. The hand device provides the concept of multiple pointers and gesture features. These pointers are directly mapped to the points of interest of the manipulation. Those composable gestures can form a base to signify various status changes.

With the above philosophy in mind, a software framework—the Virtual Panel Architecture [7]—was designed to help implement an intermediate abstraction for the manipulations of 3D objects by hand gestural input. There are three major components in the architecture (see Figure 2): the *Gesture Server* is responsible for extracting information from physical hand tracking devices and composing gestures for the use of a later stage; the *Panel Server* is in charge of maintaining a database of 3-D objects, and of reporting interactions by multiple pointers in the form of events; and the *filtering processing stage* is used to encapsulate information from the events to be sent to application programs.

SPECIFICATION OF GESTURES

In this section three basic gestures, touching, pointing, and gripping, will be discussed in the framework of the hand device and the Virtual Panel Architecture.

A gesture can be as simple as *touching*: no extra specification is needed. A gesture can be fully specified in the Gesture Server as *pointing*: here digit-oriented handshape features play the major role to define the gesture. Or, a gesture can be fully specified in the Panel Server as *gripping*: in this case the interactions of objects and pointers

are concerned. These three gestures demonstrate the usability and flexibility of our framework.

Touching

The simplest gesture is touching, that is, a 3-D pointer enters the territory of an object. It is the Panel Server's responsibility to detect the invasion of a pointer into an object, and then to report events to a filter associated with the object.

Pointing

Pointing is a gesture with a specific handshape. One of the possible ways to define pointing is as below: (1) fingers except the index one are "fully curved" and are "enclosed" by the thumb; (2) the index finger is "straight" or near straight; (3) probably, we want to restrict the orientation of pointing gesture within some range (the terms enclosed by double quotes are features in the digit-oriented handshape alphabet.). The gesture is detected by the Gesture Server if we have registered the gesture in the Server beforehand. As a result the position of the index fingertip is the starting point of the pointing; the orientation of the index fingertip is the pointing direction. Both of the values are sent to the Panel Server, which has to detect the shooting target from the fingertip information.

Gripping

Another important gesture is gripping gesture. With this gesture the click-and-drag in 2-D graphical user interface can be superseded by the hold(grip)-and-move in 3-D space.

In the beginning the concept of click is replaced by that of *holding*. A 3D holdable object has to be specified by a set of points, edges or faces which are *holdable* places on the object. When one or more fingertips and the thumb tip enter the holdable places of an object, then we regard
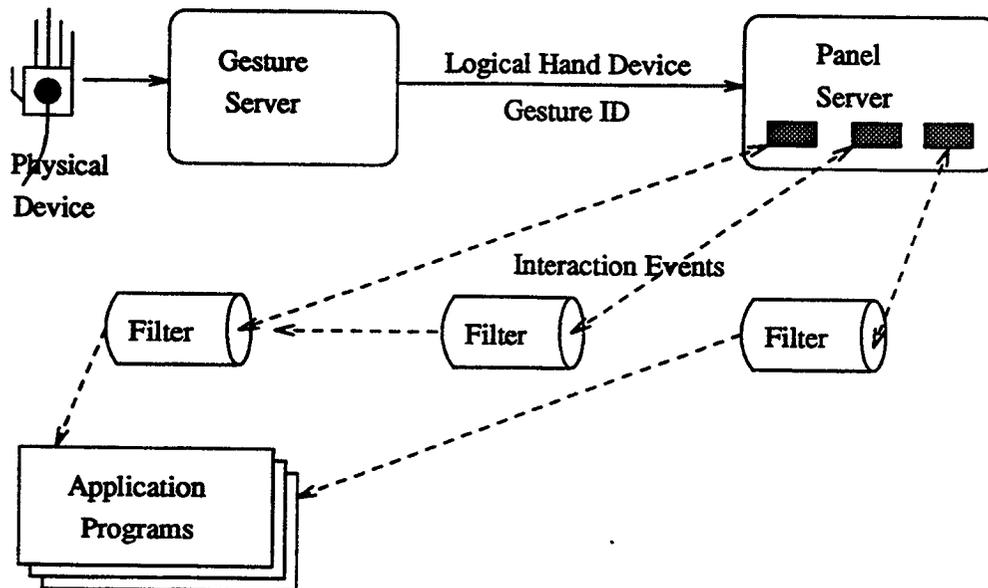
Figure 2: The Virtual Panel Architecture

the object as *being held*. The whole holding process is handled by the Panel Server, which knows that pointers are entering holdable objects. We also can release an object by letting less than two pointers stay in the holdable places of the object. As long as pointers are holding an holdable and movable object, the object can be moved around in 3D space by the hold-and-move.

An object can define its own action rules in its associated filter to react to various holdings. The *holding* can mutate with *Tip Grip*, *Pinch Grip*, *Lateral Pinch* [8], etc. to signify different states as different mouse-button combinations.

CONCLUSION

The advantages of the above user interface model in virtual environments are three-fold: the user can concentrate on limited parts of interest on the hands while the major semantics of gestural interactions are still maintained; application programmers can focus on these salient points only to simplify programming jobs; and, the computation load in the system will be relieved since the detection of precise contacts of hands upon

3D objects will be reduced from computing a whole hand into computing a number of points only.

Currently we are experimenting with the framework using a VPL DataGlove, which is connected to a Macintosh and a SPARCstation. The DataGlove does not have the power to extract all of the information on the logical hand device. However, the partial information on the hand from the DataGlove gives us a good beginning.

Our modeling of the gestures has shown that the expressive power of our user interface model is at least not less than that of a 2-D graphical user interface because of the hold-and-move operation. However, there is still a broad space in 3-D manipulation that has not been explored, especially for multi-pointer interactions. We continue the study on the model to determine if it is able to accommodate new and novel interactions. We hope this line of research will eventually benefit the standardization of 3-D manipulation in virtual environments.

REFERENCES

1. Hauptmann, A. G., and McAvinney, P., "Gestures with Speech for Graphic Manipulation", Int. J. MAN-MACHINE STUDIES, Academic Press, Vol. 38, 1993, pp. 231-249.

2. Bolt, R. A., and Herranz, E., "Two-Handed Gesture in Multi-Modal Natural Dialog", Proceedings of the ACM Symposium on User Interface Software and Technology, Monterey, California, November, 1992, pp. 7-14.

3. Baudel, T., and Beaudouin-Lafon, M., "Charade: Remote Control of Objects Using Freehand Gestures", COMMUNICATIONS of the ACM, Vol. 36, No. 7, July, 1993, pp. 28-35.

4. Figueiredo, M., Bohm, K., and Teixeira, J., "Advanced Interaction Techniques in Virtual Environments", COMPUTERS & GRAPHICS, Pergamon Press, Vol. 17, No. 6, 1993, pp. 655-661.

5. Papper, M. J. and Gigante, M. A., "Using Gestures to Control a Virtual Arm", VIRTUAL REALITY SYSTEM, Academic Press, 1993, pp. 237-246.

6. Su, S. A., and Furuta, R., "A Logical Hand Device in Virtual Environments", in VIRTUAL REALITY SOFTWARE & TECHNOLOGY: proceedings of the VRST'94 conference, Singapore, August, 1994, pp. 33-42.

7. Su, S. A., and Furuta, R., "The Virtual Panel Architecture: A 3D Gesture Framework", in the proceedings of the 1993 IEEE Virtual Reality Annual International Symposium (VRAIS), Seattle, Washington, September, 1993, pp. 387-393.

8. Kroemer, K. H. E., Kroemer, H. J., and Kroemer-Elbert, K. E., ENGINEERING PHYSIOLOGY: Bases of Human Factors/Ergonomics, 2nd ed., van Nostrand Reinhold, New York, 1990, p. 22.

Session 4
Displays