VR and Laser-Based Interaction in Virtual Environments Using a Dual-Purpose Interaction Metaphor

Frank Steinicke, Timo Ropinski, Klaus Hinrichs

WWU Münster,
Institut für Informatik,
Einsteinstraße 62,
48149 Münster, Germany
{fsteini, timo, khh}@math.uni-muenster.de

Abstract

In this paper we present a dual-purpose interaction metaphor for desktop-based and virtual reality (VR) interaction in VR system environments. Many VR interaction techniques enable users to perform 6 degrees of freedom (DOF) manipulation tasks. However, designers of most GUIs have only standard desktop devices in mind for using them. Therefore porting desktop applications to VR systems often fails since no techniques exist for making the full functionality of the GUI accessible to the user. VR interaction techniques lack support for combining both VR and desktop-based interaction, e.g., the control of the operating system and 2D menu handling. In this paper we present a dual-purpose interaction metaphor based on laser pen interaction, which is capable of performing both desktop-based and VR interaction tasks.

1 Introduction

VR environments play an important role in both research as well as industrial application areas. Because of decreasing costs and sophisticated visualization and interaction techniques more and more non-expert users have access to VR systems enabling them to interact with visualized datasets in three dimensions.

Currently adequate interaction techniques to control VR systems are usually realized by using a tracking system, which is aware of the current position and orientation of the user's input device. Thus 6 DOF interactions to manipulate the VR environment can be supported by the VR system. Problems arise when the user wants to perform actions not supported by the usually specialized VR applications. For example to switch between different VR applications requires the user to put away the currently used interaction devices and to perform the task by using a regular desktop mouse attached to the system.

In this paper we present an approach which enables users of VR systems to switch seamlessly between desktop-based and VR interactions. We propose a dual-purpose interaction metaphor, which is based on laser pen interaction [3] and enables the user to perform 6 DOF interactions as well as controlling a 2D desktop within the VR system. Thus it is possible to control graphical user interfaces, e.g., of the operating system, that have not been developed with the intention to be used in an immersive VR environment.

In the following section we will describe the dual-purpose interaction metaphor and how it can be used to perform desktop-based as well as VR interactions. In particular we will explain how the relatively small desktop widgets can be accessed using a non-2D input device. Section 3 gives implementation details regarding our integration into the Windows operating system. Section 4 presents an example application that benefits from the usage of the dual-purpose interaction approach. Section 5 concludes the paper.

2 Dual-Purpose Interaction

2.1 Advanced Laser-Based 2D Interaction

The adaptation of 3D desktop applications to VR systems often fails since no techniques are available that allow the user to access the full functionality of the application; especially menu interaction and system control is not implemented or must be completely redesigned for the usage in VR environments. Furthermore, when working with a VR system direct access to the operating system or another desktop application may be required, e.g., to use a WIMP- (window, icon, menu and pointer) based program.

Since standard input devices such as the mouse or the keyboard are popular and their use is familiar to most users, desktop-based interactions should rely on these intuitive mechanisms. With our approach 6 DOF motions and additional signals transmitted by a VR input device are mapped to the corresponding system commands to emulate mouse input events. To calculate the current mouse cursor position a virtual ray is cast from the position of the input device toward the screen, its direction depends on the input device's orientation. The intersection point between ray and screen is transmitted to the system as location for the mouse event. Thus users can handle any 6 DOF tracked input device in almost the same manner as a laser pen for desktop interaction [1].

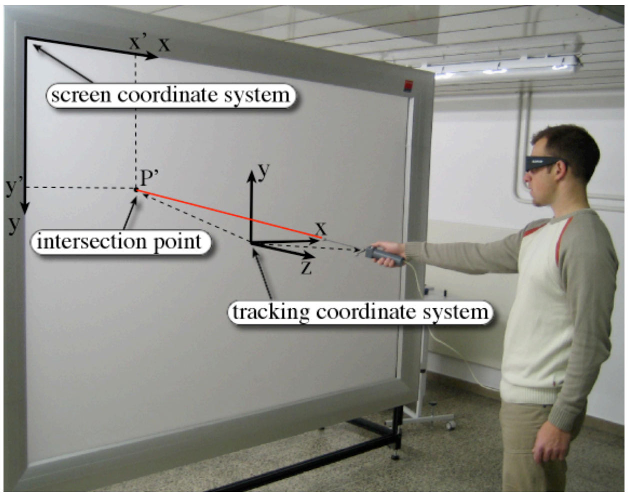

Figure 1. Laser-based interaction with a VR input device.

Figure 1 illustrates this approach. A user is placed in a VR system environment, e.g., in front of a projection wall, and interacts with the image plane using a laser-based input device.

A change of the coordinate system is necessary since the position and orientation of the input device is given in tracking system coordinates. The resulting intersection point has to be transformed to calculate the corresponding mouse position $P'=(x',y')$, and an operating system specific mouse event is generated.

The use of the described laser-based interaction with VR input devices involves various problems, such as hand tremor and difficulties when selecting of small GUI-elements, e.g., icons and widgets [3], which usually are designed for mouse interaction.

To solve these problems we exploit an interaction metaphor that simplifies the selection of objects. The Improved Virtual Pointer (IVP) metaphor enables selection of virtual objects by roughly pointing at them [2]. A ray is used to indicate the direction of the input device and an additionally visualized Beziér curve extending from the position of the input device bends to the object closest to the ray. This so-called *active object* will be chosen if the user performs a selection. The Beziér curve, which automatically bends to the active object, seems like a curved laser beam and allows aiming at virtual objects in an intuitive and natural way.

This metaphor supports the user during the selection of small GUI-elements, which are difficult to hit with the usual pointer-based metaphor. Instead of positioning the mouse cursor at the intersection point, the cursor is placed at the position of the *active widget*, i.e., the GUI-element closest to the intersection point, and sticks there until another element gets closer to the intersection point.

Mouse button events can be generated by using VR input devices, e.g., left and right mouse button or double-click events are simulated by corresponding buttons of the input device or by pinching special finger combinations with gloves.

Thus the user can control all GUI-elements that can be accessed by a desktop mouse by utilizing the 6 DOF motions of arbitrary VR input devices, and therefore any desktop-based application can be controlled with VR devices. Furthermore, special VR hardware, e.g., a projection wall and a space mouse, can be used to perform common desktop interaction tasks.

To allow desktop-based and VR interaction the user can switch between two different interaction modes: a VR interaction mode to accomplish 6 DOF selection and manipulation of virtual objects, and the above described advanced laser-based mode for desktop-based interaction. For both modes we exploit the IVP metaphor, which has been adapted for the desktop-based interaction as described above. To switch between the modes of the input device the user has to satisfy predefined conditions, e.g., pressing a special button. In addition, modes can be changed automatically depending on the system state. For example, the activation of a two-dimensional menu can switch to the desktop-based interaction mode, while a subsequent selection of a menu entry can switch back to VR interaction mode.

2.2 Haptic Input Device

To exploit the described concept of dual-purpose interaction, we have developed a VR input device that combines the following advantages:

- Handy design,
- haptic feedback,
- input support and
- wireless communication.

Because of its ergonomic design this haptic input device (see Figure 2) can be used extensively in VR environments. It weighs less than 250g, and the cylindrically shaped grip measures 155mm in length and

30mm in diameter. Thus the haptic input device is handy and user-friendly.

The passive markers attached to the bar construction on top of the grip support 6 DOF tracking via an optical tracking system without constraining the user's movements by cables. To maintain the unconfined freedom of movement a wireless connection is used for transmitting vibration level information and button events. The vibration unit is connected to a Bluetooth® module, which can be attached to the belt of the user. This module exchanges signals wirelessly with a Bluetooth® adapter connected to the serial port of the host computer.

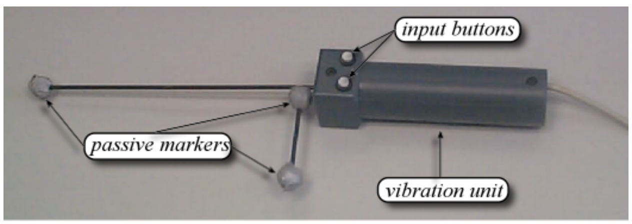

Figure 2. Haptic input device.

2.3 Multimodal Interaction

Since multimodal interaction has been proven to enhance HCI, we support multimodal interaction concepts by using the vibration unit integrated into the haptic input device. This unit can generate arbitrary haptic feedback by varying the duration of the vibration signal, the duration of the periods between subsequent signals, and the total number of signals. In addition, auditory signals can be used to support the user during the interaction process. When the active object changes, i.e., the Beziér curve bends to another object or the mouse cursor snaps to another widget, the user receives adequate auditory and haptic feedback. The position of the user and the active object are used for the sonification process as well as for the level of vibration. A gentle sound dispersing from the position of the virtual object respectively the two-dimensional GUI-element towards the user's position supports the user during the interaction. The level of intensity for the vibration and auditory feedback can be altered depending on the distance between the user and the active GUI-element. Starting from an initial minimal level of vibration, a decreasing distance between input device and active GUI-element results in a higher level of vibration.

To perform different operations the haptic input device provides two push buttons mounted on the upper side at the top of the grip allowing an easy access with the thumb, index or middle finger. One of these buttons can be used to switch between the VR and the advanced laser-based interaction mode while the other button can be used for other interaction tasks, e.g., object selection.

3 Implementation

In this section we describe implementation details of our approach. The information is given with the intention to allow porting of the concepts to arbitrary combinations of VR and graphical user interface systems.

3.1 VR Input Device Connection

A component of our VR system receives the tracking data of a VR input device, e.g., a space mouse or the previously described haptic input device, and transforms the position and orientation of the input device to the corresponding screen position. As described in Section 2.1 this data is used to generate corresponding operating system events. The tracking system generates tracking data and sends them via network to the computer which controls the dual-purpose interaction. Since this network communication is based on the user datagram protocol (UDP) which is available for most platforms, a platform independent determination of the screen intersection point is assured. Mouse events are generated by operating system specific commands; this is currently implemented for Windows platforms only and is based on the *SendInput* function of the *Windows User Interface* API.

Using these concepts the dual-purpose interaction can be integrated in a VR application and thus support VR and desktop-based interaction within the application. Alternatively, the laser-based mode of the metaphor is usable without any reference to a VR application and can be utilized to control graphical operating systems and WIMP-based desktop applications.

3.2 Determination of the Active Widgets

To exploit the advantages of the IVP metaphor the position and region of every accessible GUI-element is needed to determine the element which is closest to the screen intersection point of the ray. For each selectable GUI-element the distance to the intersection point has to be determined. On the image plane the Euclidian distance between the intersection point and the centre of each GUI-element is used as distance metric. The position and region of each active widget is inquired from the operating system using appropriate APIs and libraries. For the Windows operating system this can be achieved by using the *SendMessage* function in conjunction with appropriate parameters (*LVM_GETITEM, LVM_GETITEMPOSITION*, *LVM_GETITEMRECT* etc.).

4 Application

In cooperation with the department of city planning in Münster (Germany), we develop a residential city planning software for desktop computers. A menu supports the desktop-based interaction, among them generation, selection, manipulation and deletion of virtual buildings. Using the dual-purpose interaction metaphor the port of this application to a VR system environment requires minimal adaptation. Only an interface to the tracking system, stereoscopic rendering features and the dual-purpose interaction metaphor are added to the desktop-based application. Hence VR interactions can be performed while guaranteeing access to the complete menu originally designed for the desktop application. Figure 3 shows this application in a responsive workbench (RWB) environment.

Another benefit of the proposed concepts advances development of VR applications and systems. VR developers can modify and restart applications from within the VR system environment without leaving it, because an access to all necessary components is guaranteed.

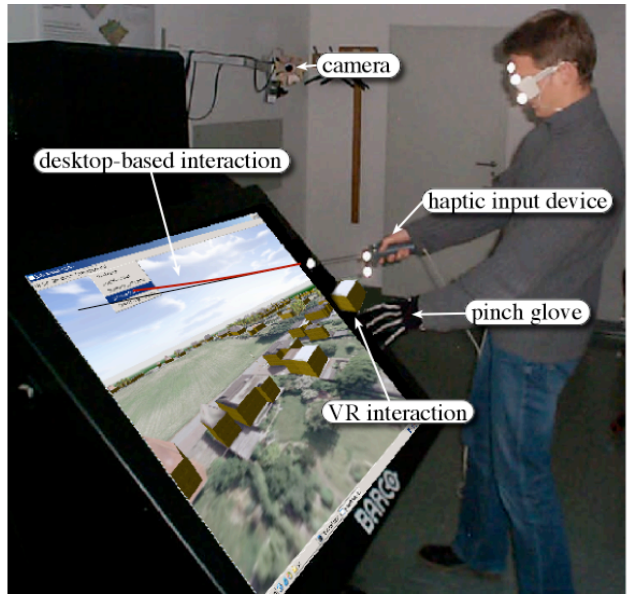

Figure 3. Desktop-based application in a RWB environment.

5 Conclusion and Future Work

In this paper we have described a novel approach for switching seamlessly between desktop-based and VR interactions in VR environments. Under this approach arbitrary applications can be controlled with interaction devices which are usually utilized to control VR environments. Furthermore the described concepts can be adapted easily to different operating systems.

By using the described dual-purpose interaction based on the IVP metaphor we have experienced a comfortable interaction with two-dimensional GUI-elements as well as with three-dimensional VR environments.

Since rotation around and translation along the laser axis does not affect the intersection point, 4 DOF are sufficient to control a laser-based metaphor.

In the future we will investigate how the remaining 2 DOF can be exploited to manage 3D desktop replacements (3D-Desktop, 3DNA, SphereXP etc.) of existing 2D desktops.

References

[1] K. Cheng and K. Pulo, “Direct Interaction with Large-Scale Display Systems using Infrared Laser tracking Devices”, In Proc. Australian Symposium on Information Visualisation, (invis.au'03) (24), pp. 67-74, 2003.

[2] F. Steinicke, T. Ropinski and K. Hinrichs, “Object Selection with an Improved Virtual Pointer Metaphor”, International Conference on Computer Vision and Graphics (ICCVG), Warsaw, Poland, 2004 (in print).

[3] M. Wissen, “Implementation of a Laser-based Interaction Technique for Projection Screens”, ERCIM News, No.46, pp. 31-32, 2001.