OBJECT SELECTION IN VIRTUAL ENVIRONMENTS USING AN IMPROVED VIRTUAL POINTER METAPHOR

Frank Steinicke, Timo Ropinski and Klaus Hinrichs
Institut für Informatik, Westfälische Wilhelms-Universität Münster, Germany

Abstract In this paper we describe object selection techniques and metaphors for virtual environments (VEs). By combining and extending known techniques, we introduce an improved virtual pointer metaphor which enhances interactive object selection. The extension of the virtual pointer metaphor is based on a bendable ray, which is visualized by a quadratic beziér curve. While a straight ray visualizes the direction of the virtual pointer, the bendable ray points to the closest selectable object. Strategies for determining this object are discussed and compared.

Keywords: Virtual Reality, Interaction Techniques, Selection Metaphors

1. INTRODUCTION

Virtual environments (VEs) have shown considerable potential as an intuitive and natural form of human-computer interfaces. Improving the acceptance of virtual reality (VR) technology requires optimization of the most basic interaction techniques to maximize user performance and provide efficient human-computer interaction.

Before interacting with virtual objects, the user needs to specify the target for the desired interaction. This *selection* is generally considered as an interaction technique itself. In this paper direct interaction metaphors for object selection in VEs are discussed. To perform an object selection a set of *selectable* objects, a technique for identifying the object to be selected and a mechanism to indicate the time of selection are required. Furthermore, the user should get an adequate feedback, e.g. visible, audible or tactile, about a possible or an already performed object selection.

Our improved virtual pointer enhances object selection techniques by combining the following advantages:

- possible selection of near as well as distant objects,

K. Wojciechowski et al. (eds.), Computer Vision and Graphics, 320–326.

- prevention of accuracy errors and ambiguities,
- sufficiency of 2 DoF for controlling the virtual input device in many application areas and
- possible selection of occluded objects.

The paper proceeds in Section 2, where we describe existing object selection metaphors. Section 3 introduces the improved virtual pointer metaphor. Applications which benefit from the use of the improved virtual pointer metaphor are described in Section 4. The paper concludes in Section 5.

2. RELATED WORK

Many basic approaches for interaction techniques in VEs have been proposed. In (Bowman and Hodges, 1997) manipulation metaphors in VEs are compared, among them the virtual hand and the virtual pointer metaphor. Both metaphors use a virtual tool, which is controlled by a real input device. They differ in the way a selection is performed.

When using the virtual hand metaphor, a selection is possible when the virtual input device intersects the desired object. For the selection of distant objects outside the immediate reach of the user, alternative strategies have to be applied. The Go-Go (Poupyrev et al., 1996) technique support distance selection by using a non-linear mapping function which translates the measured distance from the user's head to his hand into the controlled distance between the real and the virtual hand. In contrast to moving a virtual hand to the object, using virtual pointer metaphors, e.g. ray-casting techniques (Jacoby et al., 1994; Pierce et al., 1997), a selection can be performed when the virtual ray hits the desired object. In (Poupyrev et al., 1998) a comparison of the Go-Go and a simple ray-casting technique shows comparable performance for local selection conditions, i.e. selection in the immediate reach of the user, independent of the virtual object's size. With increasing distance, especially when higher selection accuracy is required, the Go-Go technique has a significant performance advantage.

By using a cone instead of a ray, as it is done in the spotlight technique (Liang and Green, 1994), accuracy errors which occur during distant selections are reduced. But more than one object may fall into the light cone. In (Forsberg et al., 1996) a modification of the spotlight technique is described, which diminishes these ambiguities by providing aperture based and resizeable selection cones.

To select fully or partially occluded objects Olwal and Feiner have described a flexible pointer (Olwal and Feiner, 2003) visualized by a curve. This approach is based on a two handed control of the curve, whereas the vector formed by the hands determines the pointer's direction. The amount of curvature is determined by the orientation of each hand.

3. THE IMPROVED VIRTUAL POINTER METAPHOR

We believe that virtual pointer metaphors are natural and require less effort for local and remote object selection. Since ray-casting techniques require a simple way to aim at virtual objects, we have developed the improved virtual pointer metaphor which avoids most disadvantages of current selection techniques.

In our approach we combine the metaphors described in Section 2 and extend them to provide an intuitive mechanism for object selection. The proposed technique allows to select the desired object without requiring an exact hit. A straight ray is used to indicate the direction of the virtual pointer, while an additionally visualized bendable ray points to the closest selectable object. This object, which would be chosen when a selection is performed, is called *active object*.

3.1 Distance Calculation

For improved object selection the object with the minimal orthogonal distance to the virtual ray has to be determined. This minimal orthogonal distance may refer to different reference points of a desired object, e.g. the center of the object's bounding box, the nearest vertex, the nearest edge etc. (De Amicis et al., 2001). If more than one object have the same minimal distance to the virtual ray, different strategies may be considered, e.g. the object closest to the user becomes active. The minimal distance is calculated by dropping a perpendicular from the reference points of all considered objects to the virtual ray. Figure 1 illustrates how the distance vector d_i, from the virtual ray to a reference point of an object obj_i, is calculated using the ray direction, the vector from the virtual pointer position to the reference point of obj_i and the angle α_{obj_i} between them.

The distance calculation has to be performed for every scene object considered. But the scene graph structure used in most computer graphic systems can be exploited to enhance the performance. Thus we almost maintain the same frame rates as achieved using the classical virtual pointer metaphor. The results of the distance calculations are stored in an ordered list, which can be utilized to switch between selectable objects, e.g. to select an occluded object by *tabbing* through the ordered list.

3.2 Region Examination

The number of considered objects depends on the scene structure and the examined region. When the user moves the virtual pointer through the VE, the object closest to the virtual ray within an appropriate region has to be determined. The optimal choice for structure and size of this region depends on the scene's configuration, i.e. the number and the arrangement of its selectable objects. We distinguish between two possible concepts: Constraining the number of considered objects depending on intersections with geometric shapes and considering all selectable objects.

In the first case only objects intersected by a predefined geometric shape are considered. Possible geometric shapes are cones, cylinders, spheres, boxes etc. which may be attached to the virtual input device or located somewhere in the scene. Those shapes are only used for calculation purpose without being visualized. If one or more selectable objects intersect such a geometric shape, the object with the smallest distance to the ray becomes the active object. If no such intersecting object exists, the geometric shape is enlarged and tested again for intersecting objects. The enlargement process is repeated until either an intersection is found or the complete scene has been examined without success, i.e. the scene does not contain any objects.

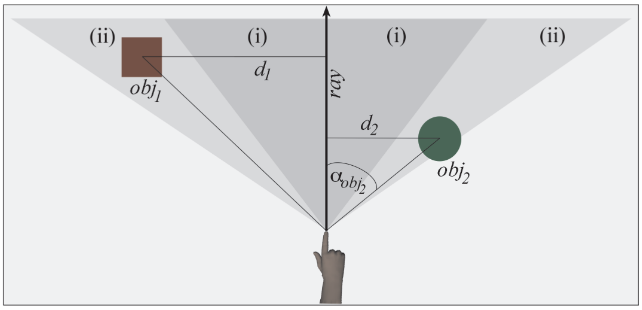

Figure 1. Example configuration with initial (i) and extended (ii) cone.

Figure 1 illustrates an example. Since the virtual ray does not hit any object, an initial cone-shaped region (i) is examined. Because no object is intersected by the initial cone a larger region is analyzed. Now that two objects are within the cone (ii) and due to $|d_2| < |d_1|$, obj_2 becomes the active object. This strategy is favorable for extremely densely populated scenes but the appropriate size of the region to be examined depends on the topology of the scene.

Alternatively, the distance for all selectable objects can be calculated. This may be advantageous in VEs in which the initial examined region usually does not contain any objects. Thus multiple examination of regions can be avoided. But this means, that the distance to the ray must be calculated in one scene traversal for all selectable objects. However, using the described distance metrics actually is not very complex, such that this approach is sufficient for most VEs.

3.3 The "Sticky-Ray" Metaphor

In a densely populated VE with large objects and small gaps between them, a different strategy may be advantageous. As in other techniques a ray is casted through the VE and the first object to be hit becomes the active object. It remains active until the virtual ray hits another selectable object. Therefore, selection is simplified because only a single hit of a desired object with the selection ray is needed to afford a selection. This approach leads to a "sticky-ray" metaphor as illustrated in Figure 2. In the beginning the ray intersects

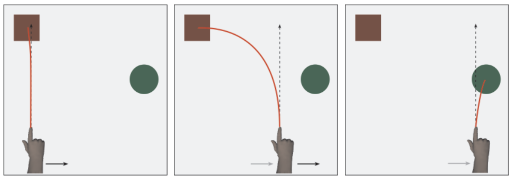

Figure 2. "Sticky-ray" example during a translation of a virtual input device from left to right.

the red box leading to a feasible selection (left). After moving to the right, the red box is still active (indicated by the red curve) although the green sphere is closer to the virtual ray (middle). Moving further to the right, the ray hits the green sphere which then becomes the active object (right). In contrast to the concepts described in the previous section the "sticky-ray" metaphor needs no distance calculation at all.

3.4 Visualization of the Virtual Ray

To get an adequate visual feedback of a possible selection, we visualize the ray direction vector as well as the position of the active object. Visualization of both aspects is ensured by visualizing, in addition to the ray direction vector, a beziér curve graph

$$B(x) = \sum_{i=0}^{2} p_i \cdot \binom{2}{i} x^i (1-x)^{2-i}, \tag{1}$$

with $x \in \mathbb{R}^3$ and three points $p_i \in \mathbb{R}^3$, $i = 0, \ldots, 2$ defining the curve. The anchor points p_0 and p_2 are defined by the position of the virtual input device and the active object's reference point, e.g. the center of its bounding box. Between these start and end points the control point p_1 is located on the ray direction vector and determines the bend of the beziér curve. Our tests have indicated that $0 < |p_0 - p_1| < |p_0 - p_2|$ has to be satisfied, otherwise the attraction is too low or appears to be unnatural. Figure 3 shows a virtual scene illustrating the attraction for two selectable objects with p_1 chosen such that $|p_0 - p_1| = \frac{4}{5} \cdot |p_0 - p_2|$.

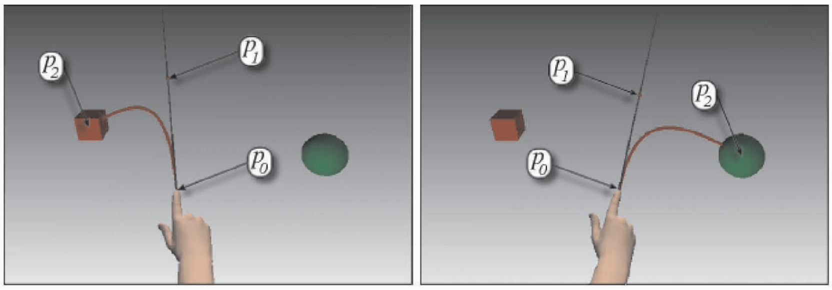

Figure 3. The red box is active although it is not hit by the virtual ray (left). After a small rotation the green sphere is located closer to the ray and becomes the active object (right).

4. APPLICATIONS

Because of their natural and intuitive usability virtual pointer metaphors are applied in many VR applications. Especially VR systems, in which distant object selection is essential, utilize ray-casting techniques and benefit from our approach. We present two application areas in which we effectively use the improved virtual pointer metaphor. We have experienced, that our approach allows advanced interaction for both, VR novices and VR experts. The two described application areas are used in a responsive workbench environment, as well as in desktop environments. The user controls the virtual pointer by either an optical tracked 6 DoF input device or a space mouse.

In Figure 4 (left) the improved virtual pointer metaphor is used in a spatial planning environment for selection and manipulation of virtual buildings. Figure 4 (right) shows the usage of the metaphor in a menu-based VE offering the user to experiment with platonic solids. In addition to the solids the virtual ray is attracted by all menu entries allowing an easy access to each menu item. In both environments interactions could be performed more efficient using the improved virtual pointer metaphor.

5. CONCLUSION AND FUTURE WORK

In this paper we have introduced an improved interaction metaphor for selection tasks in VEs. We have described its advantages and gave some sample applications. The proposed metaphors promise advanced usability and more intuitive interaction with VEs. To enhance further performance appropriate combinations of the approaches described in Section 3 can be used. Currently we are setting up a user study to evaluate our improved virtual pointer metaphor. In this study we are going to compare different strategies used to simplify the selection of occluded objects. Furthermore, we will evaluate the use of different positions for the curve's control point p_1 and how omitting

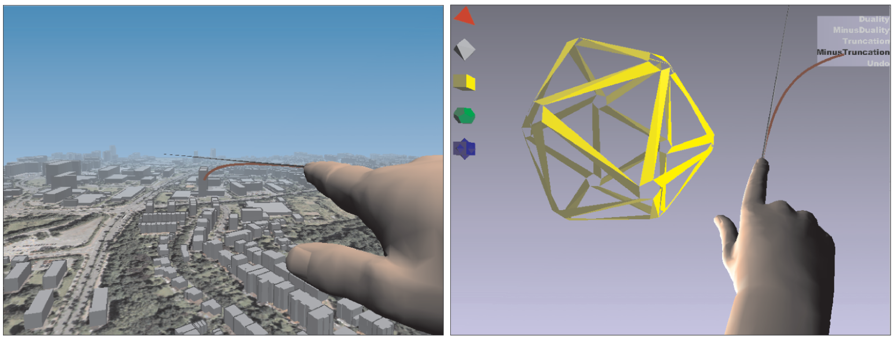

Figure 4. Interaction with a spatial planning environment (left) and a menu-based VE (right).

visualization of the straight ray, indicating the direction of the input device, affects user interaction.

REFERENCES

Bowman, D. and Hodges, L. (1997). An evaluation of techniques for grabbing and manipulating remote objects in immersive virtual environments. In *ACM Symposium on Interactive 3D Graphics*, pages 35–38.

De Amicis, R., Fiorentino, M., and Stork, A. (2001). Parametric interaction for CAD application in virtual reality environment. In *XII ADM INTERNATIONAL CONFERENCE on Design Tools and Methods in Industrial Engineering*, pages D3/43–D3/52.

Forsberg, A., Herndorn, K., and Zeleznik, R. (1996). Aperture based selection for immersive virtual environments. In *ACM Symposium on User Interface Software and Technology*, pages 95–96.

Jacoby, R., Ferneau, M., and Humphries, J. (1994). Gestural interaction in a virtual environment. In *Stereoscopic Displays and Virtual Reality Systems: The Engineering of Virtual Reality*, pages 355–364. SPIE.

Liang, J. and Green, M. (1994). JDCAD: A highly interactive 3D modeling system. *Computers & Graphics*, 18(4):499–506.

Olwal, A. and Feiner, S. (2003). The flexible pointer: An interaction technique for selection in augmented and virtual reality. In *ACM Symposium on User Interface Software and Technology (UIST Conference Supplement)*, pages 81–82.

Pierce, J., Forsberg, A., Conway, M., S. Hong, R. Zeleznik, and Mine, M. (1997). Image plane interaction techniques in 3D immersive environments. In *ACM Symposium on Interactive 3D Graphics*, pages 39–44.

Poupyrev, I., Billinghurst, M., Weghorst, S., and Ichikawa, T. (1996). The Go-Go interaction technique: Non-linear mapping for direct manipulation in VR. In *ACM Symposium on User Interface Software and Technology*, pages 79–80.

Poupyrev, I., Weghorst, S., Billinghurst, M., and Ichikawa, T. (1998). Egocentric object manipulation in virtual environments: Empirical evaluation of interaction techniques. *Computer Graphics Forum*, 17(3):41–52.