Two-Handed *Through-the-Lens*-Techniques for Navigation in Virtual Environments

Stanislav L. Stoev[1], Dieter Schmalstieg[2], and Wolfgang Straßer[1]

[1] WSI/GRIS, University of Tübingen,
Auf der Morgenstelle 10, C9,
72076 Tuebingen, Germany
{sstoev, strasser}@gris.uni-tuebingen.de
[2] Vienna University of Technology
Favoritenstrasse 9-11
A-1040 Vienna, Austria
dieter@cg.tuwien.ac.at

Abstract. We present a tool set of techniques for navigation in virtual environments. Based on a *through-the-lens* concept, the user is enabled to simultaneously view the surrounding virtual world and an arbitrary location of the virtual world as seen through an interactively defined window. For the manipulation and the adjustment of this virtual window we propose three different strategies, loosely based on the well-known *eyeball-in-hand*, *scene-in-hand*, and *world-in-miniature* techniques. Since our technique provides a preview area with controllable size and position, the occluded part of the scene can be flexibly managed, while enabling intuitive and precise work with the proposed tools. The proposed techniques provide powerful tools that are easy to implement and can be applied in any type of virtual environments.

1 Introduction and Related Work

One of the most important features defining usability and user acceptance of a virtual environment, is the support of adequate *navigation*. Very large virtual environments make efficient navigation a key requirement. The *navigation* can be divided in three groups of techniques [12]: *Searching* is the motion to a particular location in the virtual environment. *Exploration* is defined as navigation without particular target. Finally, *maneuvering* is the high-precision adjustment of the user position in order to perform other tasks. Beside the application of these navigation techniques for performing particular tasks, each of them has a different application range. Searching and exploration techniques are utilized for overcoming large distances, while maneuvering is applied rather locally. In this work, we will present three tools each of which is suitable for accomplishing one or more of these tasks.

Various navigation techniques belonging to the above navigation types are described in the literature. Darken and Sibert [3] propose 2D maps for navigation, applying principles extracted from real world navigation aids. Stoakley et al. [10] extended this idea into the third dimension. The authors define navigation as a term covering two related tasks: movement through a virtual space and determining the orientation relative to the surrounding environment. Stoakley et al. [10] introduce in their work the *World In Miniature* (WIM)-technique for manipulating objects in the 3D space. Pausch et al. [5] extended the WIM concept to provide a navigation tool, enabling the user to directly manipulate the current viewpoint for searching and exploration tasks. However, they also reported that despite the intuitive application of the WIM, the direct viewpoint manipulation was confusing to many users. This is caused by the fact that the world surrounding the user moves simultaneously with the movement performed on the WIM. In addition, due to the applied miniaturized fixed-size copy of the virtual world, precise manipulation and navigation tasks are difficult to perform.

Another set of navigation tools is discussed by Ware and Osborne [14], who evaluate three metaphors for exploration and virtual camera control: *eyeball-in-hand*, *scene-in-hand*, and *flying-vehicle-control*. They conclude: "None of the techniques is judged the best in all situations, rather the different metaphors each have advantages and disadvantages depending on the particular task". Similarly to the WIM-technique, the main problem with the *eyeball-in-hand* and the *scene-in-hand* techniques is that the viewpoint is directly manipulated and the resulting image immediately displayed. This, however, may confuse the user or even cause loss of orientation.

The navigation tools we present in this work are inspired by the *eyeball-in-hand*, *scene-in-hand* (we call *grab-and-drag*), and WIM-techniques, but attempt to overcome their limitations. We introduced to these tools a *through-the-lens* (TTL) concept, extending their functionality and improving their usability. The idea of this concept is based on a more generalized 3D magic lens [1, 13], therefore *through-the-lens* concept. In particular, we allow the scene to be viewed from an additional (second) viewpoint, whereas the image rendered from this second viewpoint is displayed in a dedicated window. In this way, the presented navigation aids provide a set of flexible and powerful tools, covering all of the navigation categories introduced above.

The remainder of this paper is organized as follows. First, we introduce the idea of the TTL-concept, and discuss the requirements for the realization. In Section 3, we will describe in detail the three proposed tools. We will discuss their implementation, utilization, and show snapshots of the system in action. Finally, we will conclude the paper with a short comparison of the presented navigation aids.

2 Through-the-Lens Concept

The main idea of a TTL-tool is to provide an additional viewpoint and display the scene as seen from this viewpoint in a dedicated viewing window (or output window W_o) as shown in Figures 1 and 2. Thus, a preview window is provided. This window is mapped onto a pad held in the non-dominant hand of the user. For the realization of the proposed tools, we utilized the *Personal Interaction Panel* (*PIP*) concept [11, 8]. The PIP consists of a tracked palette, on which the virtual tools are displayed is such a way, that the user sees them on the pad's surface. This is accomplished either using back-projection and a transparent palette (Virtual Table, Virtual Workbench, Cave, Powerwall etc.) or front-projection (opaque and transparent head-mounted-displays). The pen is a manipulation tool for interacting with the pad. It is a physical object with a virtual counterpart used to manipulate the virtual tools on the pad.

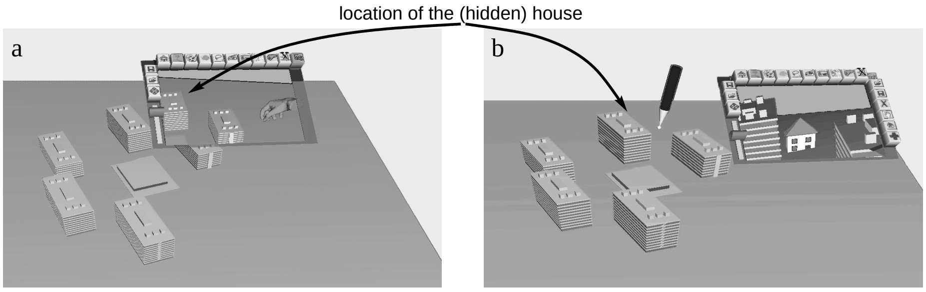

Fig. 1. The grab-and-drag tool is shown in action (Image (a)). After manipulating the additional viewpoint (shown in (b)), details invisible from the current user position become visible. The pen shows the location of the small house in the surrounding scene. Even though, the small house is in the surrounding environment, it is not visible from the current user position.

Considering the additional viewpoint and the scene seen through it (called secondary scene/world), there are two conceptually different states of the pad and the second world viewed through the window W_o on the pad: Either the secondary world is fixed in the space, or it is fixed with respect to the viewing window. In the first case, the output window W_o mapped on the pad is used to explore parts of the secondary scene (which is fixed in the space) using a magic lens (see Figure 2(c) and (d)). The window can be freely moved in the secondary world's coordinate space. In contrast, in the second scenario the window can be adjusted to show a given part of the secondary scene (location of interest), such that even if the output window W_o is moved, the virtual window W_v remains fixed in the secondary world's space (see Figure 2 (c) and (e)). Throughout this work, we will refer to these two states as "secondary scene fixed in space" and "window fixed in secondary scene". The latter scenario is only used when the TTL-WIM tool is applied, as will be shown later.

As introduced above, the proposed concept of providing an additional viewpoint for an explored scene is a more generalized 3D magic lens [1, 13] and extends the *seam* concept (spatially extended anchor mechanism) described in [9]. In contrast to the original *seam* implementation, however, in our scenario the scene viewed through the virtual window and the window itself are not fixed with respect to each other. We provide a dynamic viewpoint for exploring distant locations in the virtual world. Thus, an interactively defined *wormhole* is introduced, allowing viewing the currently explored world from an arbitrary remote location. One of the most important features of the proposed through-the-lens concept is that the surrounding virtual world remains unchanged during the two-handed manipulation of the additional viewpoint. Depending on the applied tool, the second viewpoint can be either manipulated directly (*eyeball-in-hand*) or indirectly (TTL-WIM and TTL-*grab-and-drag*), as will be discussed later.

Finally, the proposed tools are used not only to view the scene from a different point of view, but also to immerse the secondary scene through this viewpoint. Thus, a controlled "teleporting" is performed. In order to enter the secondary world as seen

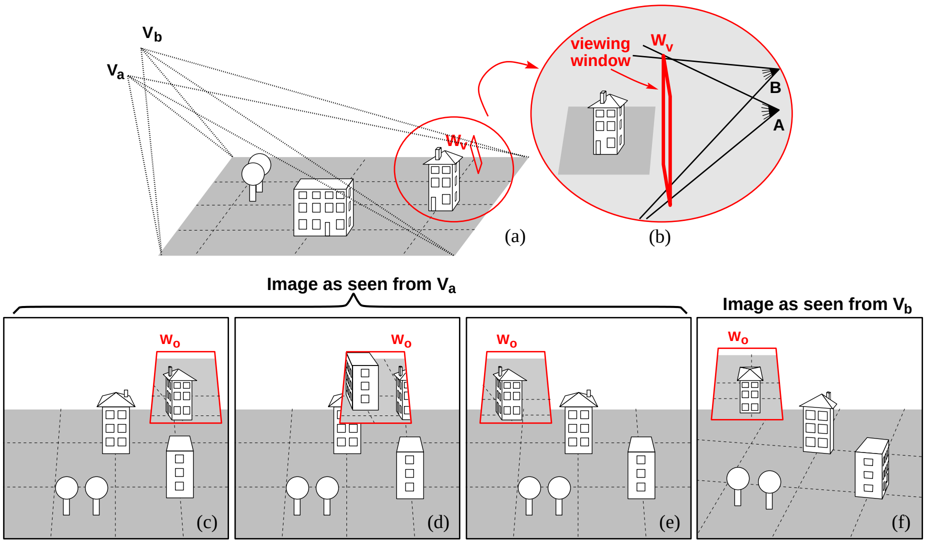

Fig. 2. The position of the virtual window (W_v) is fixed with respect to the scene seen through it, as shown in (a). V_a and V_b are two different viewing positions. (b) shows the two viewing positions A and B, derived from the current camera positions V_a and V_b. In case the virtual window W_v is fixed in the scene and the output window W_o is moved in the scene surrounding the user, the scene seen through W_o moves with the viewing window as show in (c) and (e). (d) shows the "scene fixed in space" scenario with the scene frozen as shown in (c). Moving W_o (compare (c) and (d)) allows viewing different parts of the scene. When the viewpoint changes (e.g. from V_a to V_b), the scene show in W_o can be viewed from different angles as show in (e) and (f).

from the additional viewpoint, the user has to move the pad towards the face until the image on the pad completely covers the viewing area. To facilitate the navigation, the secondary world can be immersed only if it is "fixed in the space". This is the more natural way of entering a distant world.

2.1 Realization

The hardware realization of the proposed tools requires a virtual environment setup with three tracked devices: One for the user's head position and orientation and one for each hand. This is the only information we need in order to realize the tools described in this work.

Assuming that the virtual world is organized in a scene graph-like structure, the software realization is simple as well. In order to display the virtual world as seen from an additional viewpoint V', the entire scene S, preceded by an appropriate transformation (T_a), passes the rendering pipeline once again. The projection of the output window (mapped on the pad) defines the area in which the rendering of S as seen from V' is performed. During rendering, the virtual counterpart of the physical pad is applied to fill the OpenGL's stencil buffer with an appropriate mask. Afterwards, S is rendered within the masked area as seen from V'. Further implementation details are discussed in [9].

3 TTL-Tools

As introduced above, the proposed techniques can be divided in two groups: Techniques applying direct and indirect viewpoint manipulation. In the next section, we will first describe the indirect manipulation techniques, which allow to intuitively adjust the scene seen through the second viewpoint without having to explicitly define a virtual camera position and view direction. Afterwards, we will discuss in detail the *eyeball-in-hand*-technique for direct viewpoint manipulation.

3.1 TTL Grab-and-Drag

Starting with two aligned viewpoints, this technique allows for the user to manipulate the scene seen from the additional viewpoint by grabbing and dragging the scene in the desired direction (Figure 3). This approach is similar to the scaled-world grab loco-

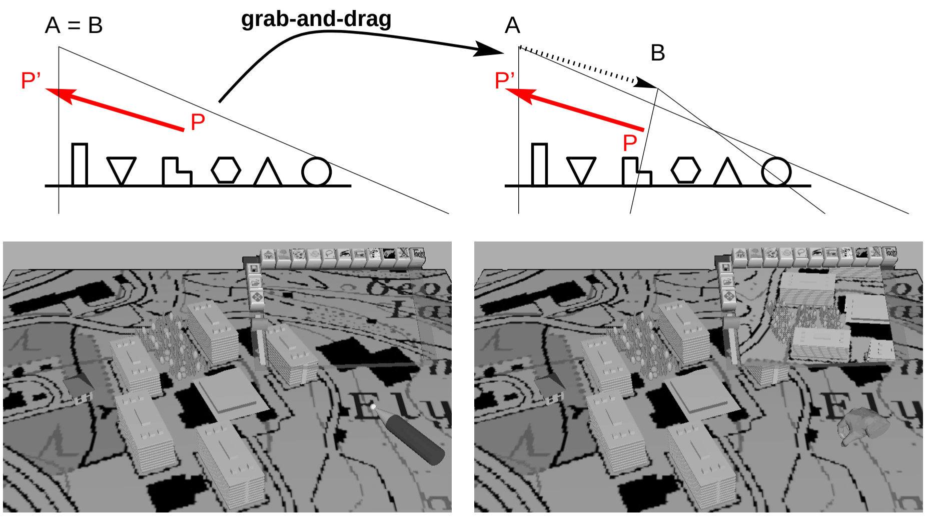

Fig. 3. Initially, both viewpoints are aligned as shown on the left. Grabbing the scene at point P and dragging to point P' corresponds to a translation (eventually combined with rotation) of viewpoint A to viewpoint B, as shown on the right.

motion metaphor described by Mine et al. [4]. Unlike the scaled-world grab where the authors apply the motion immediately, we provide a preview window. The scene seen through this window is now manipulated applying a simple grab-and-drag handle. In this way, the viewed part of the scene can be chosen very precisely, without requiring the user to physically or virtually walk/fly to the location of interest. In addition to the grab-and-drag mechanism, the user can also scale the scene if needed, hence, making this tool especially suitable for final high-precision adjustment.

Unfortunately, the application range of this technique is limited to the user's reach. When applied for viewing distant locations, the proposed technique can be circumstantial. This drawback can be overcome by combining our through-the-lens technique with other techniques for remote object grabbing (e.g. go-go technique [7], image plane techniques [6], or scaled-world grab technique [4]).

Similar to the eyeball-in-hand tool, we implemented a one-to-one mapping for the translation and the rotation originating from the grabbing tool. In this way, we can easily compute the transformation T_a. For T_a, we only need the position and orientation of the dragging handle at the beginning of the dragging motion and the current position of the handle. If another grabbing technique is applied, the transformation T_a has to be generated properly.

3.2 TTL Wim

In contrast to the original WIM tool, with the TTL-WIM we do not map the miniature copy of the virtual world *on top* of the pad. Instead, we display the latter underneath the pad's surface. Thus, the user is looking into the miniaturized virtual world through a window defined by the pad.

As introduced before, the manipulation of the additional viewpoint is indirect. The viewed part of the scene is defined by interactive dragging of a box around the target location on the pad (in the scaled down world – WIM, Figure 4). The selected part can be examined not only on the pad, but also in the virtual world surrounding the user (if visible from the current viewpoint). The size of the dragged box defines the position of the additional virtual camera, while the orientation always remains orthogonal to the virtual world (z-axis of the world) as shown in Figure 4. Thus, the virtual window is always fixed in the viewed scene.

This technique is primarily used for coarse selection of the viewed area. Once the user has adjusted the desired part of the scene to be seen through the pad, the grab-and-drag tool can be applied for precise manipulation of the additional virtual camera. In order to immerse the world seen through the output window on the pad, this world has to be "fixed in space". Since this is not the case with this tool, after selecting the location of interest, the scene has to be explicitly fixed in the space. Afterwards, the grab-and-drag tool can be used for further adjustment.

We can easily determine the scale factor and the point of interest (the middle of the dragged box), thus, the computation of T_a is simple as well. Knowing the size and location of the dragged box, the camera is positioned right over the point of interest (Figure 4).

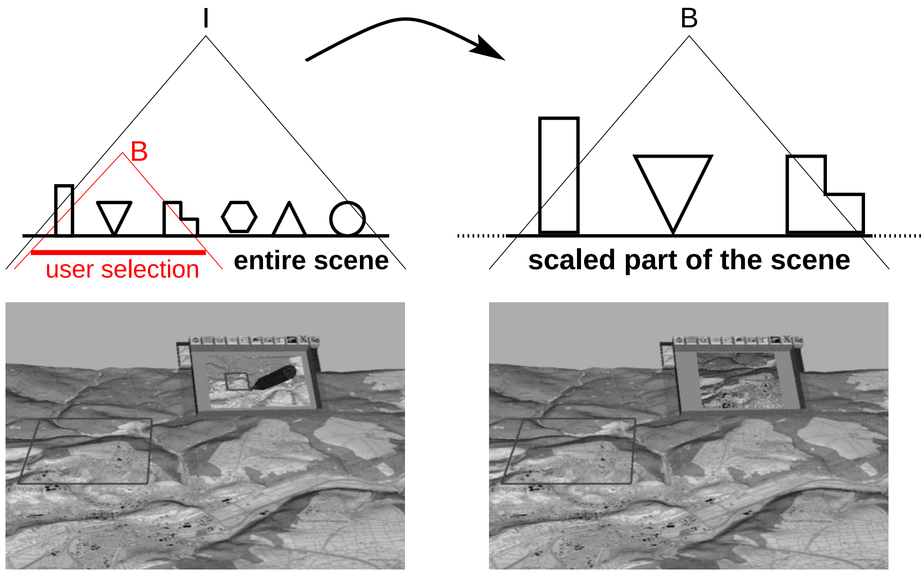

Fig. 4. Initially, a miniaturized copy of the entire scene as seen from viewpoint I is displayed on the interaction pad (do not confuse with the current viewpoint A). After interactively selecting a region of interest, the viewpoint is moved to B, such that only the selected region is visible through the pad.

3.3 TTL Eyeball-in-Hand

This technique has been introduced and explored by several researchers (e.g. [2, 14]). Despite the intuitive mental model applied with this metaphor, the main problem is the often-caused disorientation. This is due to the motion of the virtual viewpoint, which is directly applied to the current camera. Moreover, the one-to-one mapping of the hand to the virtual viewpoint makes precise adjustment of the virtual camera very hard. In addition, in order to move the camera around the scene, the user has to physically walk. Even though, the eyeball-in-hand metaphor is simple to understand and requires a simple mental model of the scene, the above limitations make it unsuitable as a sole navigation technique. In order to circumvent these limitations, while still supporting the features of this metaphor, we introduced a preview window to the eyeball-in-technique. This makes it possible to view the scene from various viewing position (in the hand's reach) without changing the current viewpoint of the user, thus, reducing confusion and disorientation.

The pen held in the dominant hand is used to define the additional viewpoint (Figure 5). It defines the position and the orientation of the new virtual camera. The scene, seen from this new viewpoint, is displayed in a window, which is mapped on the pad held in the non-dominant hand. Since the user sees the position of the virtual camera in

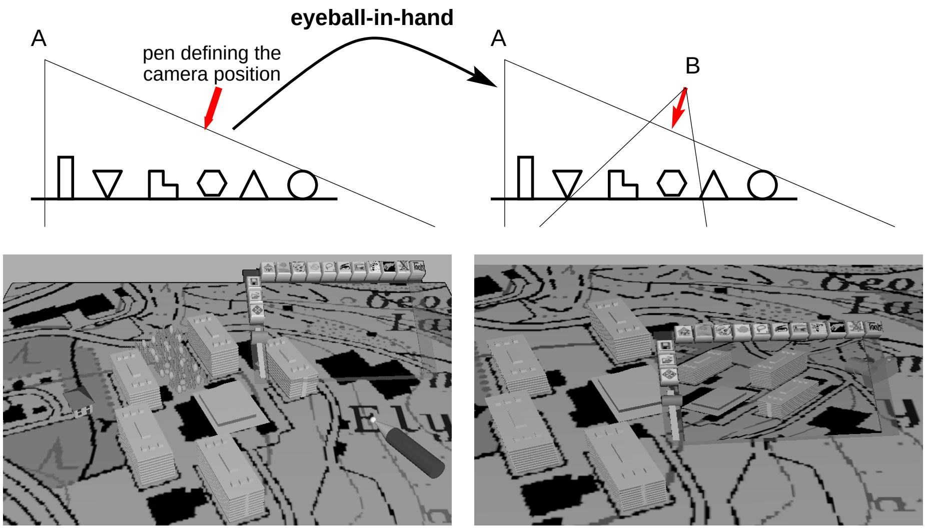

Fig. 5. Applying the eyeball-in-hand tool, the additional viewpoint can be positioned directly by defining a position and orientation of the new virtual camera B. A is the current viewpoint.

the surrounding environment and the scene as seen by the positioned camera *simultaneously*, the virtual camera can be positioned very precisely.

In order to display the appropriate part of the scene, we first compute the transformation T transforming the current position and orientation of the pen to the current camera's position and orientation. The matrix T^{-1} is now used to compute the transformation T_a.

4 Future Work and Conclusions

A topic of our future research will be the performance and evaluation of usability studies in order to make sound analysis of the proposed tools. Furthermore, we are working on an extension of the described techniques towards remote object manipulation, adjustment, and "through the wormhole" movement. Thus, we will show the versatility and flexibility of the through-the-lens display technique.

Although each of the proposed techniques has some limitations, the combination of all of them provides a powerful toolkit for exploring distant locations in a virtual world, as well as navigating in the virtual environment (see Table 1). The described through-the-lens world-in-miniature technique overcomes the main drawbacks of the original WIM-metaphor. The arbitrarily adjustable scale factor allows for the user to view large virtual worlds at a freely chosen scale. Since the miniaturized world is not displayed *on* top of the hand-held palette, the scale size is not limited to the distance between the user's head and the projection pad. The presented through-the-lens WIM is primarily used for accomplishing searching and (coarse-level) exploration tasks.

Table 1. Comparison of the proposed navigation tools.

Technique	Features	Limitations
TTL grab-and-drag	- suitable for searching tasks, and precise final adjustment tasks - intuitive viewpoint manipulation	- circumstantial for distant objects and locations
TTL WIM	- suitable for exploration and searching tasks - supports multiple scale levels	- scene cannot be entered until not "fixed in space" - improper for fine manipulations
TTL eyeball-in-hand	- requires very simple mental model - easy to use for fine precision camera adjustment	- unsuitable for exploration

In contrast, the grab-and-drag tool can be applied for all of the navigation categories introduced in Section 1. The only limitation is that it is not easily used for setting the viewpoint to very distant locations. In this case, the application of this tool has to be either preceded by the application of the through-the-lens WIM tool, or alternative techniques for remote grabbing have to be used.

The eyeball-in-hand tool is mainly used to directly define the view position and orientation in the reach of the user's hand. Hence, it falls into the category of the maneuvering-tools. This technique is very intuitive and requires a simple mental model. With the proposed preview window, this tool provides a valuable navigation aid.

In summary, we presented three tools for navigation in virtual environments based on a through-the-lens display technique. This concept allows exploring distant locations or invisible features of the virtual world without having to virtually or physically fly/walk to the remote location. It enables the viewing of locations of interest, while the world surrounding the user remains unchanged. This main contribution of the presented work will allow enhancing existing navigation aids and developing new tools exploiting the through-the-lens concept.

5 Acknowledgements

This work was partly funded by the Austrian Science Fund (FWF) under contract no. P14470-INF.

References

1. E. A. Bier, M. C. Stone, K. Pier, W. Buxton, and T. D. DeRose. Toolglass and magic lenses: The see-through interface. In *SIGGRAPH 93 Conference Proceedings*, volume 27, pages 73–80, 1993.
2. J. Brooks, F. P. Grasping reality through illusion- interactive graphics serving science. In *Proceedings of ACM CHI 88 Conference on Human Factors in Computing Systems*, pages 1–11, 1988.

3. R. P. Darken and J. L. Sibert. A toolset for navigation in virtual environments. In *Proceedings of the ACM Symposium on User Interface Software and Technology*, Virtual Reality, pages 157–165, 1993.
4. M. R. Mine, F. P. Brooks, Jr., and C. H. Séquin. Moving objects in space: Exploiting proprioception in virtual-environment interaction. In T. Whitted, editor, *SIGGRAPH 97 Conference Proceedings*, Annual Conference Series, pages 19–26. ACM SIGGRAPH, Addison Wesley, Aug. 1997. ISBN 0-89791-896-7.
5. R. Pausch, T. Burnette, D. Brockway, and M. E. Weiblen. Navigation and locomotion in virtual worlds via flight into Hand-Held miniatures. In R. Cook, editor, *SIGGRAPH 95 Conference Proceedings*, Annual Conference Series, pages 399–400. ACM SIGGRAPH, Addison Wesley, Aug. 1995.
6. J. S. Pierce, A. S. Forsberg, M. J. Conway, S. Hong, R. C. Zeleznik, and M. R. Mine. Image plane interaction techniques in 3D immersive environments. In M. Cohen and D. Zeltzer, editors, *1997 Symposium on Interactive 3D Graphics*, pages 39–44. ACM SIGGRAPH, Apr. 1997. ISBN 0-89791-884-3.
7. I. Poupyrev, M. Billinghurst, S. Weghorst, and T. Ichikawa. The go-go interaction technique: Non-linear mapping for direct manipulation in VR. In *Proceedings of the ACM Symposium on User Interface Software and Technology*, Papers: Virtual Reality (TechNote), pages 79–80, 1996.
8. D. Schmalstieg, L. M. Encarnaçáo, and Z. Szalavári. Using transparent props for interaction with the virtual table (color plate S. 232). In S. N. Spencer, editor, *Proceedings of the Conference on the 1999 Symposium on Interactive 3D Graphics*, pages 147–154, New York, Apr. 26–28 1999. ACM Press.
9. D. Schmalstieg and G. Schaufler. Sewing worlds together with SEAMS: A mechanism to construct complex virtual environments. *Presence - Teleoperators and Virtual Environments*, 8(4):449–461, Aug. 1999.
10. R. Stoakley, M. J. Conway, and R. Pausch. Virtual reality on a WIM: Interactive worlds in miniature. In *Proceedings of ACM CHI'95 Conference on Human Factors in Computing Systems*, pages 265–272, 1995.
11. Z. Szalavári and M. Gervautz. The personal interaction panel - A two handed interface for augmented reality. *Computer Graphics Forum (Proceedings of EUROGRAPHICS'97)*, 16(3):335–346, 1997.
12. A. van Dam, A. S. Forsberg, D. H. Laidlaw, J. J. LaViola, Jr., and R. M. Simpson. Immersive VR for scientific visualization: A progress report. *IEEE Computer Graphics and Applications*, 20(6):26–52, Nov./Dec. 2000.
13. J. Viega, M. J. Conway, G. Williams, and R. Pausch. 3D magic lenses. In *Proceedings of the ACM Symposium on User Interface Software and Technology*, Papers: Information Visualization, pages 51–58, 1996.
14. C. Ware and S. Osborne. Exploration and virtual camera control in virtual three dimensional environments. In *Proceedings of the 1990 Symposium on Interactive 3D Graphics, Special Issue of Computer Graphics, Vol. 24*, pages 175–183, 1990.