5 Proof of Concept

The aim of this chapter is to prove the feasibility of the rendering techniques described in chapter 4 with the aid of adequate proof-of-concept prototypes. In addition, we describe a variety of adapted VR/AR techniques (especially interaction, but also real object registration, real-virtual object occlusion and collision detection, optical/non-optical pre-distortion, etc.) and outline possible applications of projection-based Augmented Reality configurations.
Section 5.1 introduces the Reflective Pad - a hand-held full mirror that is used to interactively increase the limited viewing volume which is constrained by semi-immersive projection screens. Thus, it improves the window violation problem which occurs with such systems. The Transflective Pad that is described in section 5.2 tends to overcome the problem of occlusion of virtual objects by real ones which is linked to rear-projection systems in general. It consists of a hand-held half-silvered mirror beam-splitter that is used as an interactive optical combiner. Together with rear-projection screens, the Transflective Pad supports Augmented Reality tasks. In section 5.3, we discuss the Extended Virtual Table. This PBAR configuration consists of a stationary screen-mirror setup and uses a large beam-splitter. The goal of the Extended Virtual Table is to increase the small field of view and to decrease the error caused by tracking distortion, which are both shortcomings of the hand-held devices. A variant of the Extended Virtual Table is outlined in section 5.4. The Transflective Board represents a seamless combination of a whiteboard and an optical combiner. It addresses the flexibility problems of the Extended Virtual Table (i.e., mobility and direct-manipulative interaction) and offers a sketch-based interaction interface. Multiple users and a seamless surround view are supported by the Virtual Showcases, introduced in section 5.5. These PBAR devices apply planar multi-mirror setups or curved mirror configurations and have the same form factor as real showcases -making them compatible with traditional museum displays.
Note that the photographs shown in this chapter are not embellished. They were taken as seen from the viewer's perspective. However, the printouts may appear darker and with less luminance than in reality. All prototypes support stereoscopic rendering, but they were switched to a monoscopic mode while taking the photographs.

5.1 The Reflective Pad

This section introduces the idea of using real mirrors in combination with rear-projection systems for the purpose of interacting with and navigating through the displayed information [Bim00a]. Subsequently a derived application is described. For this, we use a hand-held planar mirror -the *Reflective Pad*- and address a fundamental problem of applying head tracking with rear-projection planes: the limited viewing volume of these environments. Furthermore, we describe the possibility of combining a Reflective Pad with a transparent one, thus introducing a complementary tool for interaction and navigation.

5.1.1 Motivation

Head tracking represents one of the most common and most intuitive methods for navigating within immersive or semi-immersive virtual environments.
Rear-projection planes are widely employed in industry and the R&D community in the form of Virtual Tables [Bar01a] or Responsive Workbenches [Tan01b], virtual walls or Powerwalls [Sil01], or even surround-screen projection systems such as CAVEs [Cru93] or CABINs [Hir97]. Applying head tracking while working with such devices might, however, lead to an unnatural clipping of large or extreme-located virtual objects, of which parts are cut off by the edges of the viewing frustum (cf. figure 5.1). This situation is called *window violation* and destroys the sense of immersion into the virtual scene and represents one fundamental problem

of those environments. Overcoming this problem requires panning and scaling techniques (triggered by pinch gestures) to transform the projected scene to a manageable size [Pol99]. A synchronous transformation of the scene to a size suitable for a continuously changing viewpoint of the observer, however, would be difficult to achieve using these techniques.

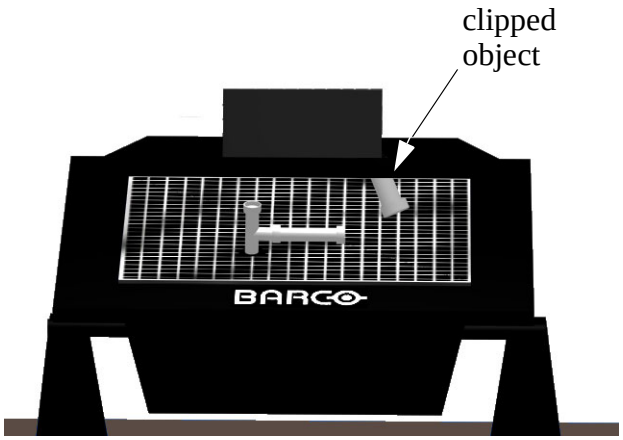


Figure 5.1: Window violation: unnatural clipping of objects.

To address this problem we propose a navigation[20] method that is complementary to head tracking. It applies a planar full mirror (the Reflective Pad) that can be used to interactively increase the perceived viewing volume of the environment.
Besides navigation, a multitude of interaction possibilities can be derived by introducing the mirror device. For example, combining the mirror device with a transparent pad leads to a particularly powerful tool that offers the application of everyday items (e.g., a pad and a pen). Thus, the fusion of the tool with the virtual world, as well as an intuitive handling, is supported.
The environment used for the following experiments is a Barco BARON Virtual Table [Bar01a]. It consists of an integrated RGB-projector that displays a 54" x 40" image on the backside of a ground glass screen. To gain the correct three-dimensional impression of the projected scene, we have applied head tracking using an electro-magnetic tracking device (e.g., an Ascension Flock of Birds [Asc01]) and stereoscopic viewing in combination with shutter-glasses (e.g., Stereographics' CrystalEyes [Ste01] or NuVision3D's 60GX [Nuv01]).

5.1.2 Increasing the Viewing Volume

A hand-held 6DOF-tracked mirror is used to complement single-user head tracking. Instead of using the physical viewpoints, their virtual reflections are applied to generate the two stereo images that enable the observer to perceive a stereoscopically and perspectively correct reflection of the projected scene in the mirror. The reflected view transform which applied in combination with the Reflective Pad is discussed in section 4.1.1. In addition to regular head tracking, the navigation possibilities are enhanced by using the mirror. Reorienting and moving it, or looking at it from different points of view allows for taking up difficult-to-reach inspection points, so that normally clipped scenes can still be observed (cf. figure 5.2).
In accordance with the term head tracking, we call this technique *mirror tracking*. In terms of providing both navigation possibilities in a complementary manner, the user can switch

20.Subsequently, we apply the term navigation to refer to a kind of camera-control mechanism in terms of gaining a new perspective of the scene, rather than to the motion of the observer.

between mirror tracking and head tracking at will (for example, by picking up and laying down the mirror). Note that mirror tracking is not a substitute for head tracking, but instead complements it.

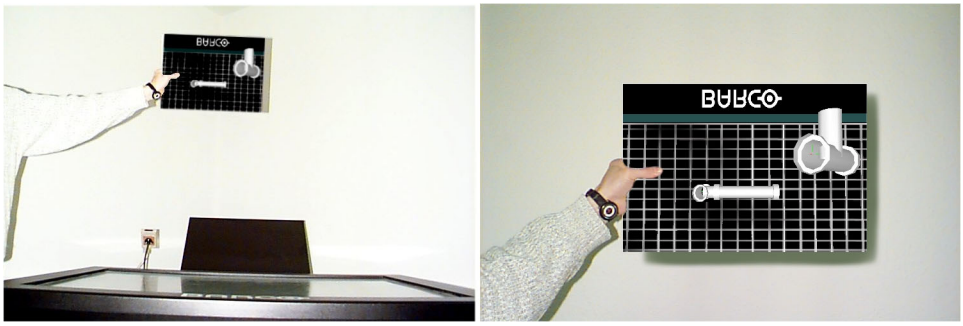


Figure 5.2: Mirror tracking with the Reflective Pad: difficult-to-reach inspection points (reflection of scene shown in figure 5.1).

Furthermore, the mirror (eventually in combination with other, fixed installed mirrors) offers the ability to distribute an infrared-signal (commonly used to synchronize the shutter-glasses with the rendering process) into different directions. This can prevent the signal from being interrupted, even if a continuous line of sight from the receiver to the sender is not given, as long as its reflection is visible in any of the mirrors.
In contrast to a manual transformation of the scene (as done in [Pol99]), mirror tracking allows us to observe unnaturally clipped areas intuitively, even when the observer's viewpoint is changing continuously.
The mirror itself can also be used as a clipping surface that enables us to slice through a virtual object and to investigate its interior. Virtual front clipping planes are attached to the mirror while being rendered together with the virtual scene. Offsetting the clipping planes from the mirror by a small amount is particularly useful for reflecting the intersection itself in the mirror.

5.1.3 Interacting with the Reflection Space

Our mirror-interaction paradigm supports indirect manipulative metaphors using ray pointers (e.g., ray casting, gaze-directed interaction and pointing [Pol99]): Instead of using the original pointing selector, the ray pointer's reflection ray must be applied to identify objects or locations in the physical space, while pointing at their reflection behind the mirror plane (cf. figure 5.3). To compute the reflection ray, equation 2.12 can be applied.

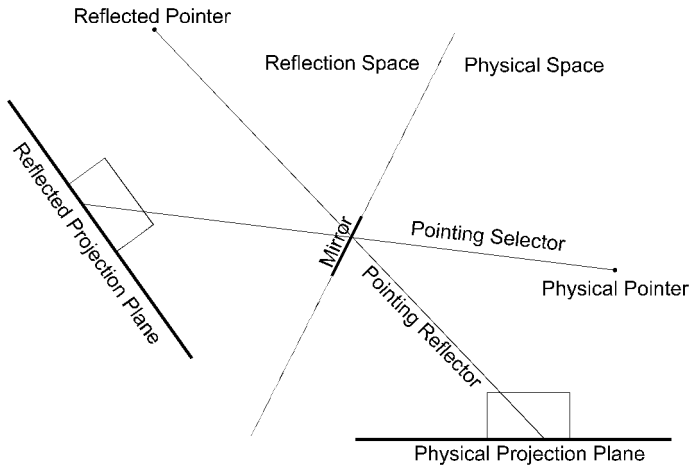


Figure 5.3: Pointing interaction with the reflection space.

This simple illustration represents the basis for all ray-pointing interactions with the reflection space. Note that a direct manipulation (such as virtual hands, direct picking [Pol99]) of the reflections is not possible because of the physical constraints of the mirror. Indirect interaction metaphors, however, require additional pointing devices.

5.1.4 Combination with the Translucent Pad

Inspired by the work of Szalavári and Gervautz [Sza97] who use opaque pen and pad props in combination with see-through head-mounted displays, Schmalstieg, Encarnação, and Szalavári [Schm99] and Coquillart et al [Coq99] apply a tracked hand-held Plexiglas pad and a pen to support multiple two-handed interaction techniques. The so-called *Translucent Pad* is augmented with 3D graphics from the Virtual Table's display and offers a wide variety of interaction possibilities. It can serve, for instance, as a palette for tools, controls and objects as well as a window-like see-through interface.

Since we have found that the application of 6DOF-tracked, hand-held transparent and reflective pads in combination with rear-projection are complementary to each other in application possibilities and interaction range, we propose their combination.

Our approach uses a Plexiglas pad that is simultaneously transparent and reflective. To achieve this, we use a semitransparent half-silvered mirror foil (such as 3M's Scotchtint P18 [3M01]). The foil's surface is identical on both sides and either reflects or transmits light, depending on its orientation to the light source (i.e., the projection plane in our case).

5.1.4.1 Active Mode Selection

The two modes of the pad are complementary and can be switched intuitively by holding it with respect to the specific task (i.e., see-through or reflective).

Whether the see-through or the reflective mode is active can be determined as follows (cf. figure 5.4).

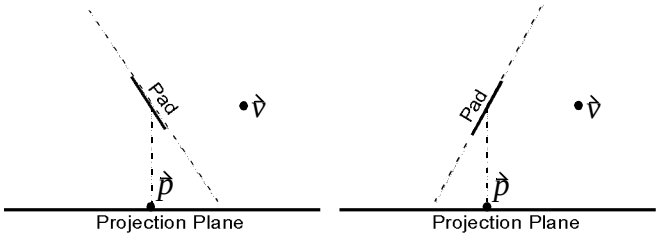


Figure 5.4: Active mode selection: division by the pad plane.

If $\vec{e}$ is the user's idealized viewpoint and $\vec{p}$ a point on the pad plane that is projected onto the projection plane, then the see-through mode is active, if

$$sign(f(\vec{e})) \neq sign(f(\vec{p})) \tag{5.1}$$

(i.e., the points are on opposite sides of the pad plane). Consequently, the reflective mode is active, if

$$sign(f(\vec{e})) = sign(f(\vec{p})) \tag{5.2}$$

where $f(x, y, z) = ax + by + cz + d = 0$ is the mathematical representation of the mirror plane.
Simply comparing whether the two points are divided by the pad plane, however, is insufficient since the reflective mode is commonly favored while holding the pad orthogonal to the projection plane, and tracking distortion makes an exact detection of the active mode impossible. Therefore, we also evaluate the solid angle between the two normal vectors of the planes and assign mode specific function zones (cf. figure 5.5).

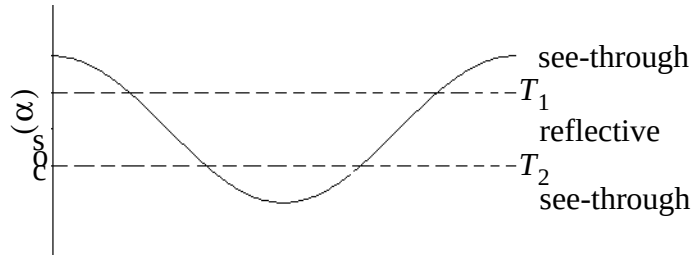


Figure 5.5: Active mode selection: function zones.

If the solid angle between the normal vector of the pad ($\vec{n}$) and the normal vector of the projection plane ($\vec{z}$) is defined as follows:

$$c = \cos(\alpha) = \frac{\vec{n}\vec{z}}{|\vec{n}||\vec{z}|} \tag{5.3}$$

then we propose the following method to determine the active mode:

$$M = \begin{cases} reflective & [(c < T_1) AND (c > T_2)] OR [sign(f(\vec{e})) = sign(f(\vec{p}))] \\ see-through & [(c \geq T_1) OR (c \leq T_2)] AND [sign(f(\vec{e})) \neq sign(f(\vec{p}))] \end{cases} \quad (5.4)$$

where T_1 and T_2 are specific threshold values that define the function zones. For our setup, we found that $T_1 = 0{,}5$ and $T_2 = -0{,}5$ are suitable values that support an intuitive switching between the modes.

5.1.4.2 Mode Functionality

Even though the modes are complementary in most cases, a certain overlap exists. On the one hand, a two-handed indirect interaction in combination with a tracked pen would be supported in the reflective mode (interaction with the reflection space), and seems to be an interesting possibility for interactions from difficult-to-reach positions (e.g., in the inside of objects, etc.). Furthermore, the application of different window-tools (such as the ones used by Schmalstieg et al [Schm99]) is imaginable in the reflective mode. On the other hand, navigation (clipping-plane-in-hand, etc.) can be realized in the see-through mode. Although this method requires overlapping application possibilities, it is still complementary in the interaction range.

Besides front-plane clipping, additional clipping planes can be found that might be set up to support rear-plane clipping from both sides of the pad plane, in both -the see-through and the reflective mode. The proper clipping plane must be activated with respect to both -the active mode and the side of the pad plane on which the user is located.

Furthermore, we can overload the see-through and the reflective modes with a multitude of different functionality. As stated in [Schm99], the user can activate different modes that are supported by the two different sides of the Translucent Pad. Thus, for instance, window-controls (such as buttons, sliders, etc.) are offered on one side, and through-the-plane tools (such as magic lenses [Bie93,Vie96], etc.) are provided on the other side. The user can switch between them anytime by turning the pad over.

The Reflective Pad, however, offers users the ability to make use of four sides (two transparent and two reflective ones), and supports a variety of application-specific interaction potentialities. Encarnação, Bimber, Schmalstieg and Chandler [Enc99], for instance, describe the possibility of using two-dimensional free-hand sketches to control and create objects by sketching on the pad.

In summary, we can assign the following major functionality to each mode:

- *Interaction in the see-through mode:* object palette, through-the-plane tools, magic-lenses [Bie93,Vie96], etc. as stated in [Schm99];
- *Complementary navigation in the reflective mode:* increase viewing volume, difficult-to-reach viewing, indirect interaction, clipping-plane-in-hand, etc.

5.1.5 Informal User Study

A Reflective Pad of the size 15" x 11" has been employed in combination with a complex virtual scenario (a 3D CAD representation of an entire ship section) to investigate its usability. Eight unbiased participants, who had never worked with or even seen the system setup were asked to explore the interior of the ship using the pad in both -the see-through and the reflective mode. The pad's functionality was explained to them and the task specified: Within two minutes, each participant had to “slice” inside the ship, navigate through its engine-room by moving the pad, and observe the asymmetric engine from all directions. After the time had expired, they were asked to sketch the engine from memory.

The goal was to find out whether or not the two different types of view (the unreflected view on the virtual scene supported by the see-through mode, and the reflected view on the virtual scene supported by the reflective mode) can be cognitively combined to achieve the correct three-dimensional plasticity, and whether the handling of the pad is intuitive.
The fact that all of the participants were able to sketch the engine rather accurately -even with most of its details at the right places-leads to the following assumption (which was verbally confirmed by the participants): Using the pad in the reflective mode while knowing that one is looking into a mirror, makes one expect to see a reflection in it. The reverse prediction can be made for the see-through mode. Thus users can cognitively combine the two types of views correctly and perceive a three-dimensional picture of the scene.
Even though the participants found the handling of the pad and the switching between see-through and reflective mode intuitive, some of them experienced difficulties with the clipping-plane-in-hand concept. This, however, can be attributed to the distortion of our electro-magnetic tracking-device (Polhemus Isotrack), which increased as soon as one moved the pad closer to the engine and consequently (in our case) further away from the transmitter.

5.1.6 Avoiding a Reflected View

Observers perceive a reflection of the virtual scene by looking at the Reflective Pad during mirror tracking. Although the informal user-study gives evidence for the assumption that observing a reflection while being aware of seeing a reflection is not critical for gaining spatial knowledge, we have extended the mirror tracking technique to avoid reflected views. Instead of generating images from the reflected viewpoints only (reflected view transform), the virtual scene is reflected over the mirror plane in addition. This reflected model-view transform is described in section 4.1.2.

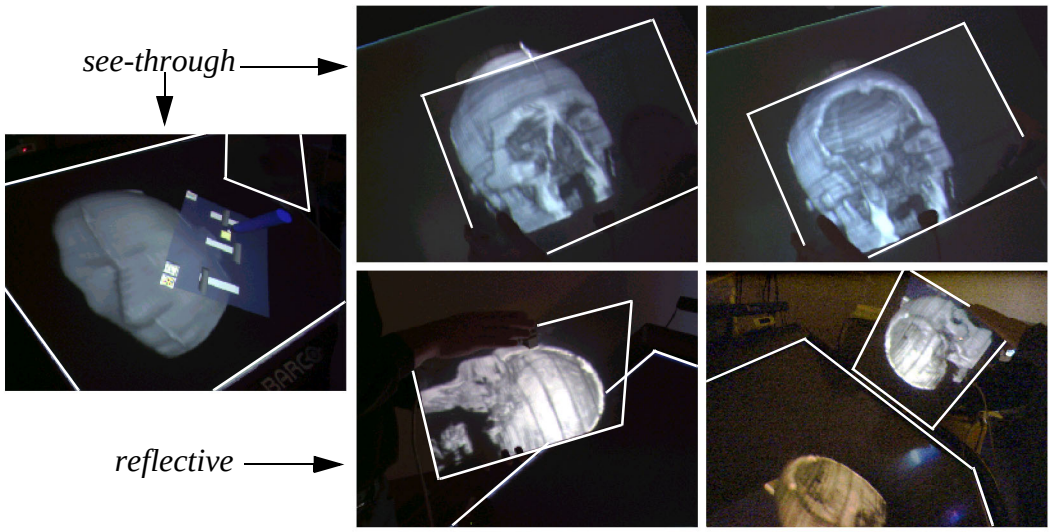


Figure 5.6: The Reflective Pad in the see-through and reflective mode.

Since, the rendered images are now reflections themselves (instead of regular perspective projections from the reflected viewpoint, only), the physical reflection of the mirror neutralizes the virtual reflection of the graphics. Consequently, the observer perceives the continuation of the scene behind the mirror in an unreflected way (i.e., just like she would perceive the perspective projection of the same scene, from the same viewpoint using the regular off-axis projection and head tracking).

Application examples for the Reflective Pad and its two modes are illustrated in figure 5.6. In the see-through mode, the pad is used to either slice computer tomographic (CT)-volumes (clipping-plane-in-hand) or to carry window controls (object palette). In the reflective mode, the pad interactively increases the viewing volume, supports to take up difficult-to-reach viewpoints and allows to slice through the volumes. The physical screen boundaries where window violations would occur and the outlines of the Reflective Pad are emphasized by white lines.

5.1.7 Discussion

We have introduced the idea of using real mirrors to view virtual worlds and have addressed the window violation problem that appears while applying head tracking in combination with rear-projection planes. A hand-held tracked mirror was used to increase the limited viewing volume of these environments.
Moving the mirror to navigate through an information space that is essentially larger than the display device (i.e., the mirror) supports a visual perception phenomenon that is known as *Parks Effect* [Par65]. That is, moving a scene on a fixed display is not the same as moving a display over a stationary scene because of the persistence of the image on the viewer's retina. Thus, if the display can be moved, the effective size of the virtual display can be larger than its physical size, and a larger image of the scene can be left on the retina. This effect can also be observed with other handheld devices, such as Fitzmaurice's Chameleon [Fit93] or the Art+Com VR Display [Art01], etc.
Since the Translucent Pad and the Reflective Pad complement each other, we proposed their combination to build a more powerful interaction and navigation tool.
We found three general advantages of the Reflective Pad:

- Users are able to handle the pad intuitively because of their familiarity with everyday physical items, such as trays, mirrors, tablets and pens, or even window controls (i.e., buttons, sliders, etc.);
- When applied with large rear-projection systems, the Reflective Pad provides a cost-efficient and easily applicable solution to combine the presented techniques with traditional (semi-)immersive VR tasks (in contrast to other approaches that make use of additional electronic tools, such as the Chameleon [Fit93], etc.);
- The Reflective Pad represents a handheld device that, in combination with rear-projection systems, offers stereoscopic viewing. To our knowledge, this cannot be realized with today's portable devices that, for instance, apply Plasma LCD screens. This is due to their low update rates.

5.2 The Transflective Pad

In this section, we describe the use of the *Transflective Pad* -a hand-held semi-transparent mirror to support augmented reality tasks with rear-projection systems [Bim00b]. This setup overcomes the problem of occlusion of virtual objects by real ones linked with such display systems. The presented approach allows an intuitive and effective application of immersive or semi-immersive virtual reality tasks and interaction techniques to an augmented surrounding space. We use the tracked mirror beam-splitter as optical combiner that merges the reflected graphics, which are displayed on the projection plane, with the transmitted image of the real environment. In our implementation, we also address traditional augmented reality problems, such as real-object registration and virtual-object occlusion.

5.2.1 Motivation

Augmented Reality (AR) [Sut65, Azu97] superimposes graphical representations of virtual objects onto the user's view of the real world. As described in chapter 3, two main display systems are commonly applied to achieve this: optical or video see-through head-mounted displays (HMD).

Optical see-through HMDs, such as Sony's Glasstron [Son00] or Virtual I/O's i-Glasses [Vir01], project the produced graphics directly in front of the viewer's eyes. Such devices always overlay the computer graphics onto the image of the real environment that is transmitted through the HMD's projection planes. The ratio of intensity of the transmitted light and the projected images can usually be adjusted electronically to integrate the images more effectively.

Video see-through HMDs use cameras to continuously capture images of the real environment. Video mixing is employed in a post-process to merge the video image with graphics on an closed-view display. Immersive HMDs, such as nVision's Datavisor family [Nvi01] generally serve as output devices to stereoscopically project the augmented images in front of the viewer's eyes.

AR-specific problems, such as the occlusion of virtual objects by real ones [Bre96], the registration of virtual objects to real ones [Whi95], or the calibration of the display system with the real environment [Jan93] are solved differently in both scenarios, thus revealing some of the distinctive advantages and disadvantages of each approach.

Large projection systems that make use of rear-projection technology are becoming more and more common within the Virtual Reality (VR) community. The introduction of real objects into virtual environments created by such systems, however, is much more difficult than the integration of virtual objects with an augmented real environment. This is due to the fact that real objects are always located between the viewer and the projection plane(s), thus always occluding the projected graphics and consequently the virtual objects. In comparison to this, the projection planes of see-through HMDs, for instance, are always located between the viewer's eyes and the real environment, and their projected graphics always occlude the real environment. An example of a VR system supporting real-object occlusion would be the projection-based virtual office [Ras98a], where front-projection is used.

The occlusion problem is the main reason for the common belief that rear-projection systems are impractical for traditional augmented reality tasks. This section introduces a possible solution to the occlusion problem using a hand-held semi-transparent mirror that allows rear-projection systems to superimpose the displayed graphical elements onto the surrounding real environment.

5.2.2 Overcoming the Occlusion Problem

As for the Reflective Pad, we used a 15" x 11" sheet of Plexiglass laminated with a semi-transparent foil to build the Transflective Pad. The foil's surface is identical on both sides and either reflects or transmits light, depending on its orientation to the light source. If viewer and light source are located on opposite sides of the plane that is spanned by the pad, the foil transmits the light, and the environment behind the pad is visible to the observer by looking through the pad. If they are located on the same side, the foil reflects the light and the viewer can observe the reflection of the environment in front of the pad.

However, if both sides of the pad are properly illuminated, the pad simultaneously transmits and reflects the light, thus both images (the reflected image and the transmitted one) are visible to the observer by looking at the pad (cf. figure 5.7). Due to these characteristics, we named our semi-transparent mirror device Transflective Pad.

We also attached a tracking emitter to the pad and used an electro-magnetic tracking device to accurately determine the position and orientation of the plane defined by the pad.

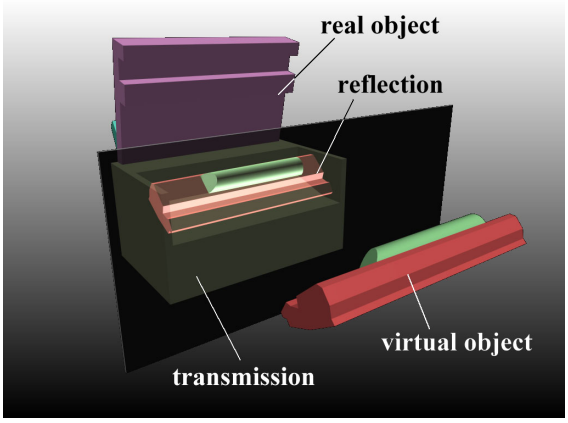


Figure 5.7: The Transflective Pad: merging real and virtual images.

The same hardware configuration that has been used for the Reflective Pad serves as environment for the following experiments. However, the projected image of the virtual scene is reflected in the mirror, while at the same time the image of the real world is transmitted through the mirror (cf. figure 5.8 for our experimental setup). Although the Reflective Pad and the Transflective Pad are physically the same, they are applied in a conceptually different way: while the Reflective Pad is either used in a reflective or in a transparent mode, the Transflective Pad is utilized in the semi-transparent (transflective = transparent + reflective) mode.
The Transflective Pad is then used as a hand-held optical combiner, whereby (in contrast to traditional rear-projection) the virtual objects always overlay the real ones. If we continuously apply the reflected model-view transform that is described in section 4.1.2, we can reorient the Transflective Pad while the registered virtual objects appear to remain in the same positions relative to the physical objects (when looking at the Transflective Pad).
The ratio of intensity of the transmitted light-the image of the physical space and the reflected light (i.e., the image within the image space)-can intuitively be adjusted by changing the angle between the Transflective Pad and the projection plane. While acute angles highlight the virtual augmentation, obtuse angles let the physical objects shine through brighter.

5.2.3 Calibration, Registration and Interaction

We use an admeasured wooden frame to calibrate the tracker's emitters to a predefined position and orientation within our global coordinate system. In addition, we apply smoothing operators and magnetic field distortion correction to the tracking samples.
While one emitter is attached to the viewer's shutter-glasses to support head-tracking, a second one is connected to the Transflective Pad to continuously determine its plane parameters. A third emitter is used to track a pen-like input device (cf. figure 5.8). Furthermore, non-stationary real objects are continuously tracked with additional emitters.

Figure 5.8: Object registration and interaction with the Transflective Pad.

To register the virtual representations to their corresponding real objects, we apply the pointer-based object registration method introduced by Whitaker et al [Whi95]. Triggered by an attached button, our pen is used to capture predefined landmark points on the real object's surface within the world coordinate system. The problem is to compute the rigid transformation between the set of predefined 3D points on the virtual representation's surface and their measured counterparts. We apply Powell's direction set method [Pre92] to solve this minimization problem and to compute the required object-to-world transformation.
Occlusion caused by real objects (static or non-stationary ones) is handled by rendering corresponding invisible (black) geometry-phantoms, as described by Breen et al [Bre96]. This model-based approach registers geometric virtual representations to real objects and renders them in black, which makes them invisible within the augmented space. If the virtual models accurately represent their real counterparts, they can be used to produce occlusion with visible virtual objects. Since both -the invisible phantoms and the visible virtual objects are reflected by the pad and overlay the real environment, the observer gets the impression that real objects, which are fully visible through their phantoms, could occlude virtual objects.
Physical objects are either placed directly on the Virtual Table's projection plane or on additional tables that surround the projection plane.
The Transflective Pad is held with one hand, while the other hand can be used to interact within the augmented physical space (e.g., with augmented physical objects -as illustrated in figure 5.11 -or with additional interaction devices, such as the pen, etc.).
We use speech commands to switch between an AR mode that is supported by the Transflective Pad, and an immersive VR mode that makes use of the Virtual Table's projection plane only. In the immersive VR mode, however, the Transflective Pad supports interaction techniques for transparent props [Schm99, Coq99] (i.e., object palette, magic lenses, through-the-plane tools, etc.) as well as Transflective Pad techniques (i.e., difficult-to-reach viewing, multiple observer viewing, clipping-plane-in-hand, etc.). These techniques are described in section 5.1.
Figures 5.9-5.11 show photographs of the transmitted physical space and the reflected projection space. The outlines of the Transflective Pad are emphasized by white lines.

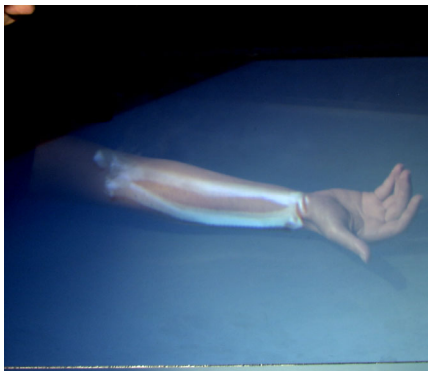


Figure 5.9: X-Ray: Overlaying forearm-bones.

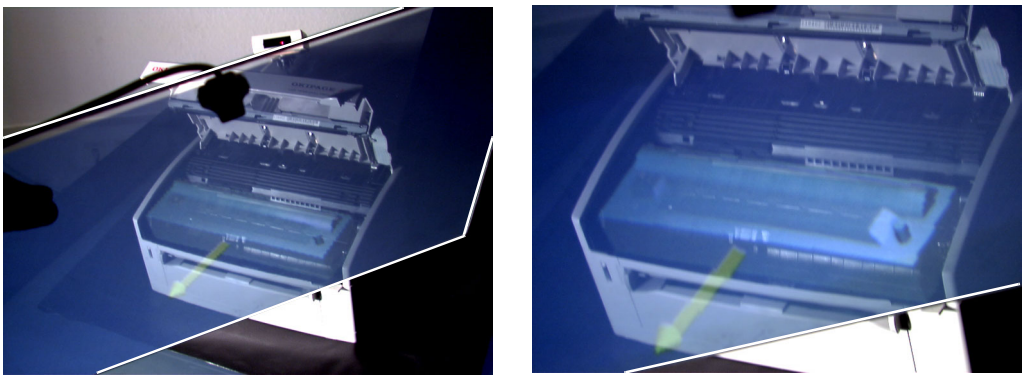


Figure 5.10: Real printer augmented with a virtual cartridge.

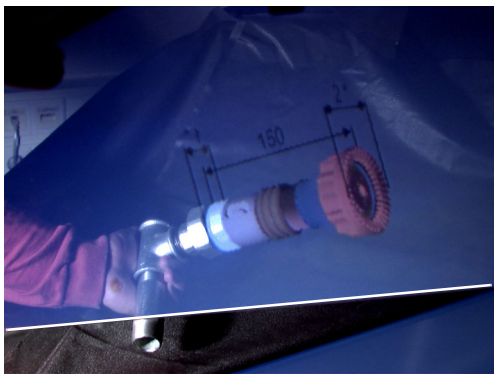


Figure 5.11: Real work piece complemented with a measured construction drawing of an extension.

5.2.4 Discussion

We have described the possibility of supporting augmented reality tasks with rear-projection systems by avoiding the occlusion problem that is linked with such display systems. A semi-transparent hand-held mirror (the Transflective Pad) is applied to fold the optical path between the projection plane and the observer and to merge the reflected graphics with the transmitted image of the surrounding real environment. Using the Transflective Pad as an interactive optical combiner that is located between the observer and the real environment, the reflected graphics always overlay the real space and make traditional augmented reality tasks possible with rear-projection systems.

The Transflective Pad offers a cheap and easily applicable way of combining the advantages of immersive and semi-immersive VR environments with less obtrusive AR techniques. It extends the viewing space and interaction space of such environments towards the surrounding real world. Those spaces would otherwise be limited by the corresponding projection planes. Especially in combination with workbench-like projection systems in which the user normally focuses on the workbench's surface, the surrounding environment can be intuitively incorporated into the interaction. As an example, consider mechanical design tasks, which can be performed on the immersive workbench and can then be evaluated by projecting the virtual model or part onto the corresponding real assembly.
However, there are several limitations compared to traditional AR systems. One of them is the indirect line-of-sight problem with the projection plane (i.e., a portion of the projection plane must be visible by looking at the mirror) that influences the application range of the Transflective Pad. This, however, is well manageable in combination with workbench-like display systems that are used to augment the close-by surrounding environment. Beside workbench-like projection systems, the Transflective Pad has also been used within a 5-sided CAVE environment. Due to the immersive characteristics of surround screen projection displays, such as CAVEs, the application range of the Transflective Pad was less restricted than with workbenches.
The pad’s small field of view is a drawback that is due to the optical combiner’s limited size. Although larger sized pads would increase the field-of-view characteristic, they would, in return, deteriorate the ergonomic properties.
Another drawback is the fact that the reflection computations (and consequently the augmentation) have to rely on at least two tracker sensors -the one attached to the pad, and the one attached to the observer's head. In contrast to single sensor setups (e.g., as see-through HMDs), time lag, nonlinear distortions and noise are slightly higher in our approach. These problems have to be compensated (e.g., by using precise tracking technology or by applying additional software solutions) to support a high-quality augmented reality system.
Note that compared to traditional AR approaches that apply HMDs, projection-based AR setups lack in mobility, which, however, does not represent a conflict with their VR-focused applications.
The proposed setup is especially versatile when interaction techniques for reflective and transparent tools [Schm99, Enc99, Coq99, Bim00b] are complemented by the possibility of supporting augmented reality for real objects.

5.3 The Extended Virtual Table

In this section, we describe a prototype of an optical extension for table-like rear-projection systems -the *Extended Virtual Table* [Bim01e]. A large half-silvered mirror is used as the optical combiner to unify a virtual and a real workbench, whereby the shortcomings that are related to the Reflective Pad and Transflective Pad (i.e., indirect line-of-sight, limited field of view, and tracking distortion) are reduced. The virtual workbench has been enabled to display computer graphics beyond its projection boundaries and to combine virtual environments with the adjacent real world. A variety of techniques is described that allow indirect interaction with virtual objects through the mirror.
Systems, such as the Extended Virtual Table, approach a conceptual and technical extension of traditional Virtual Reality by means of Augmented Reality (xVR), and a seamless integration of such technology into habitual work environments.
Furthermore, optical distortions that are caused by the half-silvered mirror combiner and the projector, as well as non-optical distortions caused by the tracking device are analyzed and appropriate compensation methods are described.

5.3.1 Motivation

Virtual Reality (VR) attempts to provide to the user a sense of spatial presence (visual, auditory, or tactile) inside computer-generated synthetic environments. Opaque head-mounted displays (HMDs) and surround-screen (spatially immersive) displays such as CAVEs [Cru93] or CABINs [Hir97], and domed displays [Ben01] are VR devices that surround the viewer with graphics by filling the user's field of view. To achieve this kind of immersion, however, these devices encapsulate the user from the real world, thus making it difficult or even impossible in many cases to combine them with habitual work environments.

Other, less immersive display technology is more promising to support a seamless integration of VR into everyday workplaces. Table-like display devices such as Virtual Tables [Bar01a, Bar01b] or Responsive Workbenches [Kru94, Kru95] and wall-like projection systems such as, e.g., Powerwalls [Sil01] allow the user to simultaneously perceive the surrounding real world while working with a virtual environment.

UNC's "Office of the Future Vision" [Ras98a] is a consequent extension of this concept. Here, in contrast to embedding special display devices into the real work environment, an office is envisioned where the ceiling lights are replaced by cameras and projectors that continuously scan the office environment and project computer graphics to spatially immersive displays. These displays could, in effect, be almost anything (e.g., walls, tables, cupboards) or anywhere in the office. While the cameras acquire the geometry of the office items (irregular surfaces), the rendering is modified to project graphics onto these surfaces in a way that looks correct and undistorted to an observer. This concept can offer both -a high degree of immersion and the integration of VR into the workspace.

Due to currently employed display technology, a main limitation of VR is that virtual environments usually cannot be mixed with the real world. If rear-projection systems are employed, real-world objects are always located between the observer and the projection plane, thus occluding the projected graphics and consequently the virtual environment. Transparent physical objects such as the transparent interaction tools described in [Coq99, Enc99, Schm99] represent an exception, since the rear-projected graphics can be used to augment such transparent props without causing major occlusion problems. If front-projection is used, physical models can be augmented with graphics by seamlessly projecting directly onto the surface of those objects instead of displaying them in the viewer's visual field [Ras98b, Ras98c, Ras99]. However, as discussed in chapter 3, this so-called spatially-Augmented Reality (SAR) concept is mostly limited to visualization and not suitable for advanced interaction with virtual and augmented real objects. Moreover, shadows that are cast by the physical objects or the user represent fundamental problems in SAR systems.

In general, Augmented Reality (AR) superimposes computer-generated graphics onto the user's view of the real world. In contrast to VR, AR allows virtual and real objects to coexist within the same space. Closed view video see-through, or transparent optical see-through head-mounted displays are currently the two main display devices for AR.

Similar to VR, the display technology that is employed for AR introduces a number of shortcomings (as discussed in chapter 3): In the case of HMDs, display characteristics (e.g., resolution, field of view, focal-length, etc.) and ergonomic factors usually interfere. While the resolution of both HMD types (closed-view and see-through) is generally low (lower than projection-based VR display devices), optical, see-through systems additionally lack image brilliance. This is because the brightness of the displayed graphics strongly depends on the lighting conditions of the surrounding real environment. Although higher-resolution see-through HMDs do exist (e.g., [Kai01]), they are often heavy and expensive. However, more ergonomic HMDs lack their optical properties.

Even if some researchers refer to AR as a variation of VR, e.g., [Azu97], a strong separation between AR and VR applications does exist, which, in our opinion, is caused mainly by the technologically constrained usage of different display devices.
In this section, we want to introduce a prototype of a cost-effective and simple-to-realize optical extension for single-sided or multiple-sided (i.e., L-shaped) table-like projection systems. A large half-silvered mirror beam-splitter is applied to extend both -viewing and interaction space beyond the projection boundaries of such devices. The beam-splitter allows an extension of exclusively virtual environments and enables these VR display devices to support Augmented Reality tasks. Consequently, our prototype supports a combination of VR and AR. As already mentioned, we refer to this combination as extended Virtual Reality (xVR), because our approach achieves a technical and conceptual extension of traditional VR by means of AR. Since table-like display devices can easily be integrated into habitual work-environments, the extension allows the linkage of a virtual with a real work-place, e.g., a table-like projection system with a neighboring real workbench.

5.3.2 Seamlessly integrating xVR into habitual Workplaces

In this subsection, we first introduce our prototype from both points of view, its physical setup and its general functioning. After a review of existing basic interaction methods for table-like display systems, we describe how virtual objects can be exchanged between both sides of the mirror and how indirect interaction with virtual objects as well as the registration of real objects can be supported through the mirror. Finally, we analyze different sources of optical and non-optical distortion that are typical for our mirror extension and describe possible compensation and correction methods.

5.3.2.1 Physical Arrangement

Our Extended Virtual Table prototype consists of a virtual and a real workbench (cf. figure 5.12).
A Barco BARON [Bar01a] serves as display device that projects 54" x 40" stereoscopic images with a resolution of 1280 x 1024 (or optionally 1600 x 1200/2) pixels on the backside of a horizontally arranged ground glass screen. Shutter glasses such as Stereographics' CrystalEyes [Ste01] or NuVision3D's 60GX [Nuv01] are used to separate the stereo-images for both eyes and make stereoscopic viewing possible. In addition, an electro-magnetic tracking device (Ascension's Flock of Birds [Asc01]) is used to support head-tracking and tracking of spatial input devices (a pen and a pad). An Onyx InfiniteReality2, which renders the graphics is connected (via a TCP/IP intranet) to three additional PCs that perform speech-recognition, speech-synthesis, gesture-recognition, and optical tracking.

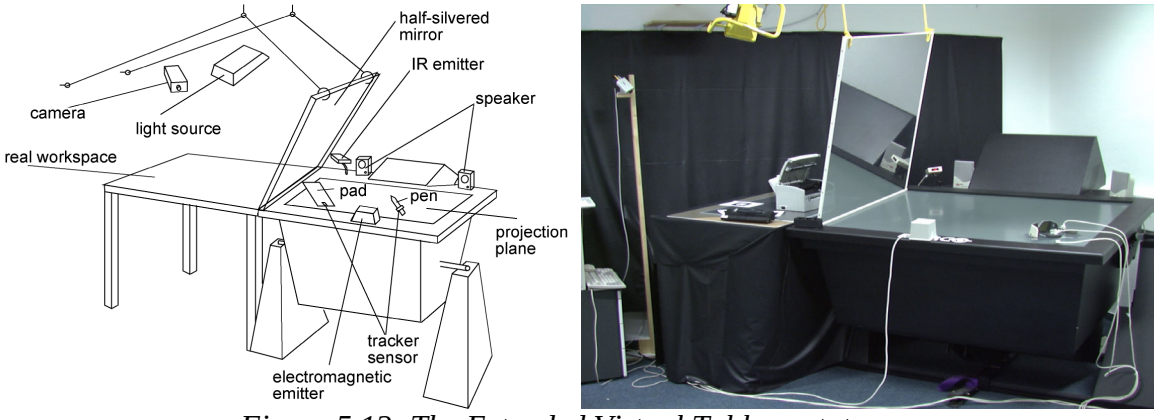


Figure 5.12: The Extended Virtual Table prototype: conceptual sketch (left) and photograph (right).

A 40" x 40" and 10 mm thick panel of glass separates the virtual workbench (i.e., the Virtual Table) from the real workspace. It has been laminated with a half-silvered mirror foil 3M's Scotchtint P-18 [3M01] on the side that faces the projection plane, making it behave like a front-surface mirror that reflects the displayed graphics. We have chosen a thick float-glass material to keep optical distortion caused by bending of the mirror or irregularities in the glass small. Our half-silvered mirror foil, which is normally applied to reduce window glare, reflects 38% and transmits 40% light[21]. Higher-quality half-silvered mirrors with better optical characteristics could be used instead (see [Edm01] for example).
With the bottom part leaning on the projection plane, the mirror is held by two strings which are attached to the ceiling. The length of the strings can be adjusted to change the angle between the mirror and the projection plane, or to allow an adaptation to the Virtual Table's slope.
A light-source is adjusted so that it illuminates the real workbench, but does not shine on the projection plane.
In addition, the real workbench and the walls behind it were covered with a black awning to absorb light, that otherwise would be diffused by the wall-paper beneath it and would cause visual conflicts if the mirror was used in a see-through mode.
Finally, a camera, a Videum VO [Win01] is applied to continuously capture a video-stream of the real workspace, supporting an optical tracking of paper-markers above the real workbench.

5.3.2.2 General Functioning

Users can either work with real objects above the real workbench or with virtual objects above the virtual workbench.
Elements of the virtual environment, that are displayed on the projection plane, are spatially defined within a single world-coordinate system that exceeds the boundaries of the projection plane, covering also the real workspace[22].
The mirror plane splits this virtual environment into two parts that cannot be simultaneously visible to the user. This is due to the fact that only one part can be displayed on the projection plane. We analyze the user's gaze to support an intuitive visual extension of the visible virtual environment. If, on the one hand, the user is looking at the projection plane, the part of the environment that is located over the virtual workbench is displayed. If, on the other hand, the user is looking at the mirror, the part of the environment located over the real workbench is transformed, displayed and reflected so that it appears as a continuation of the other part in the mirror. Using the information from the head tracker, the user's gaze is approximated by computing the single line of sight that originates at her point of view and points towards her viewing direction. The plane the user is looking at (i.e., projection plane or mirror plane) is the one that is first intersected by this line of sight. If the user is looking at neither plane, no intersection can be determined, and nothing needs to be rendered at all.
In case the user is looking at the mirror, the reflected model-view transform (discussed in section 4.1.2) is applied. Consequently, the part of the virtual environment behind the mirror is transformed in such a way that, if displayed and reflected, it appears stereoscopically and perspectively correct at the right place behind the mirror.
As for the Transflective Pad, the projected virtual environment will not appear as reflection in the mirror if we inversely reflect the graphical content from the viewer's averting side of the mirror to the opposite side and render it from the viewpoint that is reflected vice versa. The user then sees the same scene that she would perceive without the mirror if the projection plane

21.Values valid if used with 6 mm thick regular glass.
22.In our case the origin of the coordinate system is located at the center of the projection plane.

were large enough to visualize the entire environment. This is due to the neutralization of the computed inverse reflection by the physical reflection of the mirror.

The mirror-plane parameters $[a, b, c, d]$ of the mirror can be determined within the world coordinate system in different ways:

- The electro-magnetic tracking device[23] can be used to support a three-point calibration of the mirror plane;
- The optical tracking system can be applied to recognize markers that are temporarily or permanently attached to the mirror;
- Since the resting points of the mirror on the projection plane are known and do not change, its angle can be measured using a simple ruler.

Note that all three methods can introduce calibration errors, either caused by tracking distortion (electro-magnetic or optical) or caused by human inaccuracy. Our experiments have shown that the optical method is the most precise and less vulnerable to errors. Intersense's tracking system, for instance, is based on inertial navigation and ultrasound time-of-flight. It constitutes an alternative to magnetic tracking as it doesn't suffer from distortions caused by presence of metal in the environment [Fox98].

To avoid visual conflicts between the projection and its corresponding reflection -especially for areas of the virtual environment whose projections are close to the mirror- we optionally render a clipping plane that exactly matches the mirror plane (i.e., with the same plane parameters $[a, b, c, d]$). Visual conflicts arise if virtual objects spatially intersect the side of the user's viewing frustum that is adjacent to the mirror, since in this case the objects projection optically merges into its reflection in the mirror. The clipping plane culls away the part of the virtual environment that the user is not looking at (i.e., we reverse the direction of the clipping plane, depending on the viewer's gaze while maintaining its position). The result is a small gap between the mirror and the outer edges of the viewing frustum in which no graphics are visualized. This gap helps to differentiate between projection and reflection and, consequently, avoids visual conflicts. Yet, it does not allow virtual objects that are located over the real workbench to reach through the mirror. We can optionally activate or deactivate the clipping plane for situations where no or minor visual conflicts between reflection and projection occur to support a seamless transition between both spaces.

23.The tracking device has to be calibrated to the world coordinate system in advance.

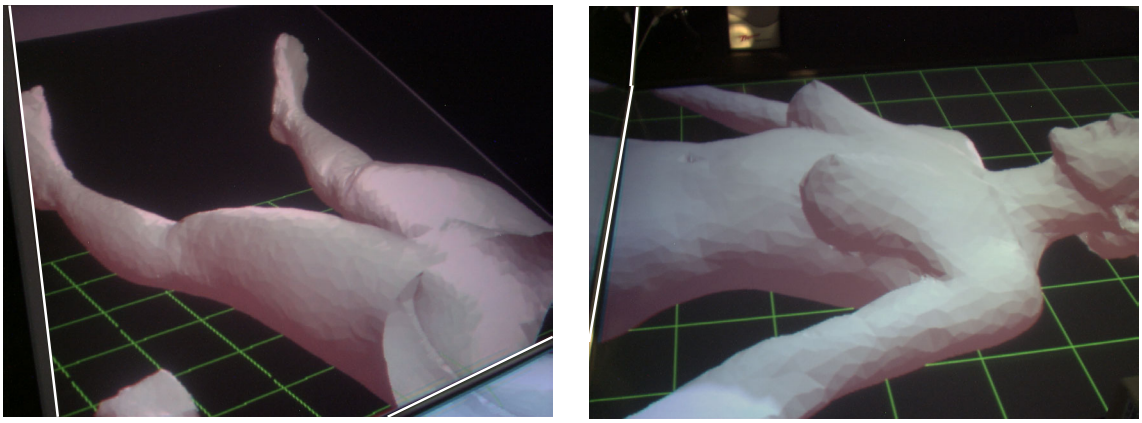


Figure 5.13: A large coherent virtual content (a life-size human body for medical training) viewed in the mirror (left), or on the projection plane (right). The real workspace behind the mirror is not illuminated.

If the real workspace behind the mirror is not illuminated, the mirror acts like a full mirror and supports a non-simultaneous visual extension of an exclusively virtual environment (i.e., both parts of the environment cannot be seen at the same time). Figure 5.13 shows a large coherent virtual scene whose parts can be separately observed by either looking at the mirror or at the projection plane.

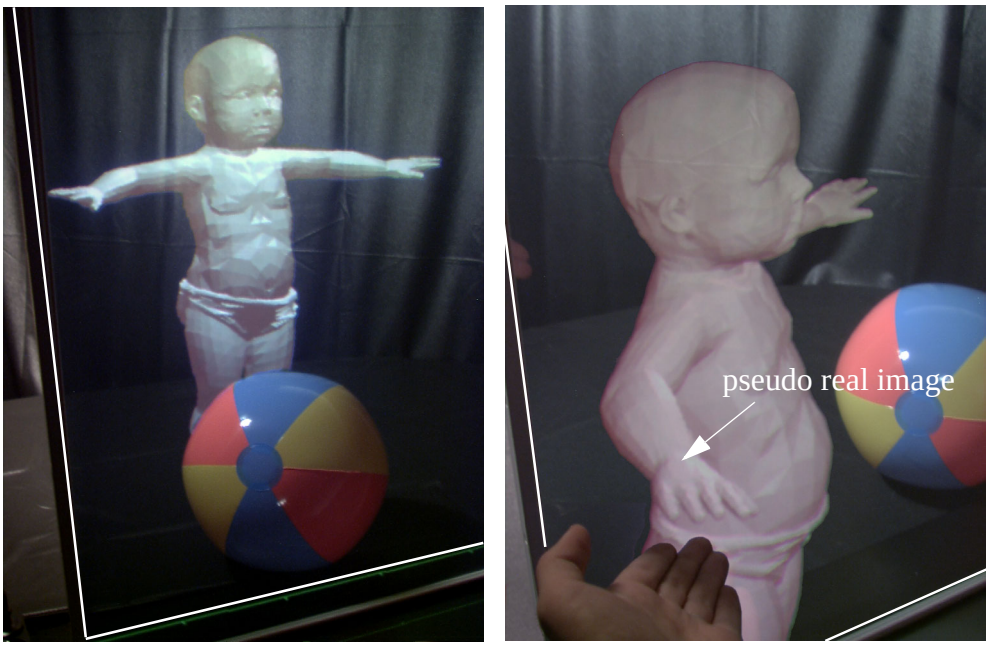


Figure 5.14: Left: Real objects behind the mirror (the ball) are illuminated and augmented with virtual objects (the baby). The angle between mirror and projection plane is 60°. Right: Without attaching a clipping plane to the mirror, the baby can reach its arm through the mirror. The angle between mirror and projection plane is 80°.

Figure 5.14 shows a simple example in which the mirror is used as an optical combiner. The outlines of the mirror are emphasized by white lines. If the real workspace is illuminated, both -the real and the virtual environment are visible to the user, and real and virtual objects can be combined in AR-manner.

Note that the ratio of intensity of the transmitted light and the reflected light depends on the angle between mirror and projection plane. While acute angles highlight the virtual content, obtuse angles let the physical objects shine through brighter[24].
An interesting optical effect can be observed by applying mirrors in combination with stereoscopic projection screens: Convex or planar mirrors can optically only generate virtual images (see chapter 2). However, in combination with a stereoscopic projection and the effects caused by stereopsis, virtual objects (i.e., the graphics) can appear in front of the mirror optics (cf. figure 5.14-right). We can refer to this effect as *pseudo real images*. In nature, real images of reflected real objects can only be generated with concave mirrors. Note that a restricted direct manipulative interaction with pseudo real images in front of the mirror optics is supported.

5.3.3 Interacting through the Mirror

In this subsection, we want to give examples of interaction techniques that are usable in combination with our Virtual Table extension. We mainly focus on methods that can be applied through the mirror -i.e., on indirect or remote interaction.

5.3.3.1 Basic Interaction Methods on table-like Projection Systems

A large variety of interaction techniques has been explored for table-like projection systems. Most of them were adopted from other output paradigms, such as HMDs, or different front/rear projection systems. Van de Pol, Ribarsky and Hodges [Pol99] present a good classification and evaluation of interaction techniques for such devices. They classify interaction into three universal tasks: *navigation* (changing position and orientation of the viewpoint), *selection* (identifying objects), and *manipulation* (changing certain parameters of objects).
Beside different environment panning and scaling techniques, head-tracking is probably the most natural (and most often applied) navigation method for table-like projection devices.
For object selection, Van de Pol identifies five different methods: *direct picking* (objects are directly picked by touching/grabbing them with the input device), *ray casting* (objects can be selected by intersecting them with a virtual laser beam that shoots out of the input device), *gaze-directed* (objects are selected by looking at them), *pointing* (objects that are pointed at, are selected), *virtual hands* (virtual hands mimic the movements of the user's real hands, but can extend faster and further, if the user reaches out to pick an object). While direct picking can only be applied to objects that are located within arm’s reach, the other methods are used for objects that are not.
Similarly, manipulation methods are divided into *direct* and *indirect manipulation*, and into *one-handed* and *two-handed manipulation*. Indirect manipulation methods (i.e., for objects that are out of arm’s reach) are classified into close manipulation (bring the object close to the user, then manipulate it directly), distance manipulation (manipulate an object at a distance), and tele-manipulation (manipulate distant objects as if they were close to the user's body - e.g., with virtual hands).
Objects that are virtually located on the tabletop are usually within arm’s reach, thus direct interaction methods (i.e., selection and manipulation) can be used. However, if they are located below the table surface, the projection plane represents a physical barrier for the user's hands, thus indirect methods have to be applied. This is also the case for our mirror extension. If objects are located behind the mirror, the mirror plane represents a physical barrier for the user's hands (and the input devices) that does not allow a direct interaction. Note that if no clipping plane is attached to the mirror, a direct interaction with objects that reach through the mirror (or are even located in front of it) is possible.

24.This is similar to the electronic shuttering of HMDs.

In this subsection, we want to focus on indirect methods that are suited to interact behind the mirror for both -exclusively virtual and augmented real environments. Note that direct interaction methods, which are applied above the virtual workbench, are not discussed in this section.

5.3.3.2 Exchanging Objects

A tracked pen is our main input device that supports direct interaction in front of the mirror and indirect interaction with objects behind the mirror. In addition, a transparent pad or tablet (which is also tracked) is applied to offer two-handed interaction as described in [Schm99, Enc99, Coq99].
Virtual objects can be exchanged between both sides of the mirror in different ways, in the see-through mode as well as in the opaque mode. For example, they can be picked with the pen, either directly or from a distance over the virtual workbench, or indirectly over the real workbench. Virtual Objects can then be pushed or pulled through the mirror (cf. figure 5.15). Note that we do not have to directly touch the virtual objects with the input device. We can rather pick them from a distance and can push/pull them through the mirror in several steps, or use a non-linear arm-extension method, e.g., Go-Go [Pop96]. Otherwise, pushing and pulling would be difficult, since the mirror represents a physical barrier for the user's hands and the input devices.

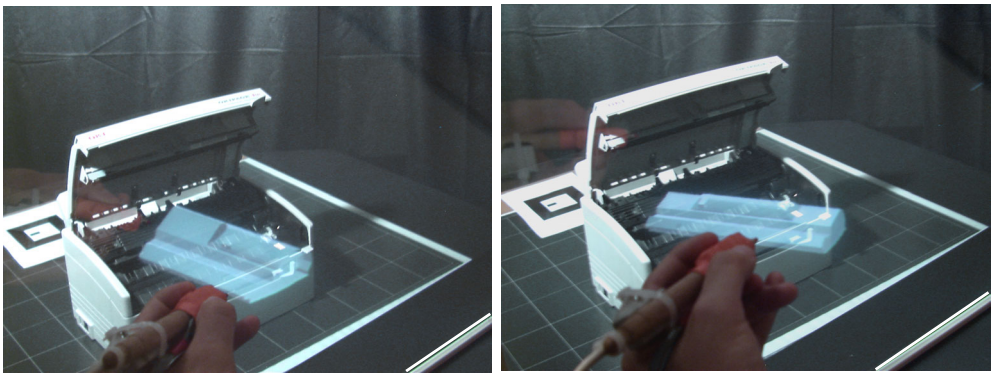


Figure 5.15: A virtual object is pushed through the mirror (a clipping plane is attached to the mirror).

Remember, that the part of the virtual environment that has to be rendered is still determined by evaluating the user's gaze, and that a clipping plane can be used to cull away the other part of the scene that does not need to be rendered. Thus, if the user picks an object and keeps on looking at it, while pushing/pulling it through the mirror, it will always be visible to her if no clipping plane has been attached to the mirror (with possible visual conflicts). If a clipping plane is attached and activated, the object -or parts of it- might not be visible while it is pushed/pulled through the small gap between the mirror and the side of the user's viewing frustum that is adjacent to the mirror -but visual conflicts are avoided. If a virtual object is located behind the mirror, an indirect interaction method has to be applied.
Another way of exchanging objects is to select them on one side, and then automatically transform them to a corresponding location on the other side. This method is further described in subsection 5.3.3.5.

5.3.3.3 Ray-Casting and Optical Tracking

As illustrated in figure 5.16, a virtual laser beam that is cast from the pen through the mirror is used to select and manipulate (i.e., to move and place) virtual objects behind the mirror plane.

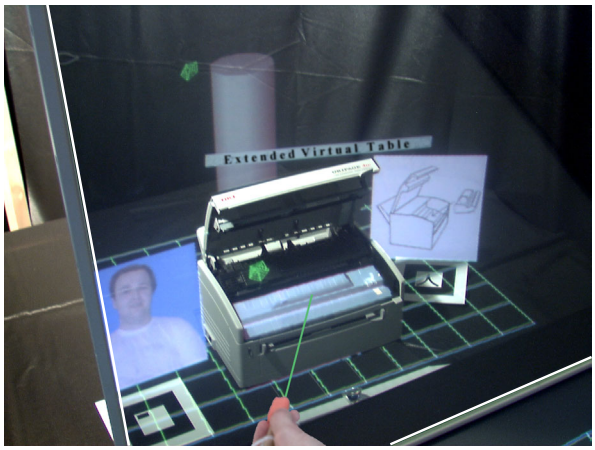


Figure 5.16: Ray-casting and optical tracking within an augmented real environment. A virtual cartridge is mounted into a real printer.

Once selected, the object follows the ray at a constant distance to the pen and can be translated and rotated in sync to the pen's movements. In addition, selected objects can be "beamed" back and forth along the ray to offer ergonomic object placement. Picking and beaming are triggered by pressing corresponding buttons on the pen.
Virtual-virtual and real-virtual object collision detection is applied over the real and the virtual workbench to simulate a realistic interaction behavior between objects. In the assembling scenario that is illustrated in figure 5.16, for example, the virtual cartridge can be mounted into the real printer under similar constraints as in reality -it cants until it slips into the casting. This fundamental collision detection capability enables us to implement gravity-based automatic object placement methods, as described by Breen, Whitaker, Rose and Tuceryan [Bre96], where virtual objects can be dropped and then fall in the direction of gravity, until they collide with other objects.
Real objects can also occlude virtual ones, as figures 5.14 and 5.16 show. This is achieved by implementing a method for see-through optical combination introduced by Breen et al [Bre96], where virtual placeholders (so-called occluders or phantoms) are registered to their real counterparts. After registration, the geometry of the occluder matches the dimensions of the corresponding real object. To make them invisible to the viewer, they are rendered in black. However, the occluder's geometry information is written into the depth-buffer and virtual objects behind it are culled during the rendering process. To avoid annoying artifacts caused by slight registration errors between occluders and real objects, the occluders are blurred to provide a smooth transition of its edges. Object-based blurring (blurring the object's geometry before rendering) or image-based blurring (blurring of the rendered image by some kind of convolution operation) can be used as described by Fuhrmann, Hesina, Faure and Gervautz [Fuh99a]. Note that the occluders also support real-virtual object collision detection, since collisions between them and other virtual objects can be detected.
Real object occlusion and collision detection is implemented for static (not movable), and for dynamic (movable) real-world objects whose positions and orientations are tracked.
Optical tracking is applied in addition above the real workbench. A camera captures a single video-stream of the real workspace and a software package [Kat99] detects different paper markers within the environment. After matching a threshold image with pre-trained patterns of the markers to be detected, the position and orientation of the identified markers are determined from their locations within the image and are then transformed into our world coordinate system. We use the markers to track real world objects, for calibrating the setup, or -as

illustrated in figure 5.16- as placeholders for multi-media information (e.g., images, video or textual information). Since different multi-media contents are attached to specific markers, they can easily be exchanged by replacing the markers, or simply be moved on the tabletop real workbench, if they are in the way. Because of performance reasons, the optical tracking is executed on an additional PC while the tracking data is sent to the rendering host via a TCP/IP connection.

5.3.3.4 Pointing and Object-Registration

Instead of using the position and orientation of the pen to determine the selector ray, as it is the case for ray-casting, the pen's position and the user's eye position can be used to compute an appropriate selector. The ray results from the two points (the pen's tip and the user's viewpoint) that span a straight line in space and supports the selection of objects that are pointed at. While obtaining the user's viewpoint from the head-tracker and the pen's position from the tracker that is attached to it, the pointing selector can be seen as a single line of sight that adjusts an object over the pen's tip.
Pointing is used to select and to manipulate objects as described in section 5.3.3.3. In addition, we found that pointing is a useful tool to register stationary real objects, whose locations are needed for collision detection and occlusion.

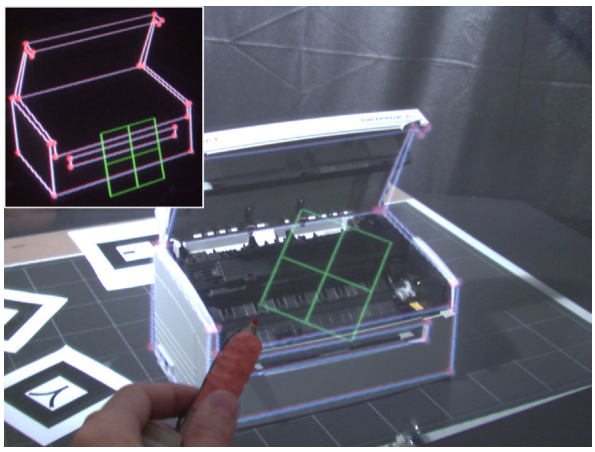


Figure 5.17: Registering a real object using pointing:
The reconstructed geometry (upper left) is overlaid over the real object.

Whitaker, Crampton, Breen, Tuceryan and Rose [Whi95] describe a pointer-based object registration method for see-through optical combination, where 3D landmark points are sampled with a spatial input device (e.g., a tracked pen) by directly touching pre-defined points on the real objects' surfaces with the pointer's tip. Then, a rigid object-to-world transformation is computed by minimizing the distance function between the sampled landmark points and their corresponding counterparts on the objects' virtual representations (i.e., their occluders). However, sampling 3D landmark points on the real objects' surfaces is inconvenient in our case, since the mirror does not allow a direct interaction within the augmented real space. Additionally, non-linear tracking-distortion over an extensive area makes a precise direct measurement of landmark points on the real object's surface impossible.
Instead, we can use pointing to support object registration. For this, we attach virtual cross wires to the pen's tip and render them monoscopically to allow a precise adjustment of points on the real object's surface with the cross wires (as illustrated in figure 5.17). If a surface point is adjusted, the selector that is spanned between the cross wires and the user's eye is computed (this is triggered by pressing a button on the pen). Since a single ray is not sufficient enough to

specify a 3D point in space, the user has to adjust the same surface point again -but from a different perspective (i.e., after moving her head). The resulting two selectors are used to compute the point in space that is closest to both rays (since they do not necessarily have to intersect). This point's coordinates are sufficient to represent the coordinates of the corresponding surface point and are used (in combination with other measured surface points) to register the virtual representation to the real object. Strictly speaking, we apply Powell's direction set method [Pre92] to minimize the distance function between measurements and predefined corresponding points.
For a parameter representation of the selectors, the point that is closest to both rays can be computed analytically as follows:

$$\lambda = \frac{(\vec{w} \times \vec{w}') \cdot ((\vec{v}' - \vec{v}) \times \vec{w}')}{|\vec{w} \times \vec{w}'|^2} \;,\; \lambda' = \frac{(\vec{w} \times \vec{w}') \cdot ((\vec{v}' - \vec{v}) \times \vec{w})}{|\vec{w} \times \vec{w}'|^2} \tag{5.5}$$

for the selectors (in parameter representation):

$$\vec{u} = \vec{v} + \lambda \vec{w} \;,\; \vec{u}' = \vec{v}' + \lambda' \vec{w}' \tag{5.6}$$

the closest point between the selectors then is:

$$\vec{p} = \frac{1}{2}(\vec{u} + \vec{u}') \tag{5.7}$$

Note that optical distortion is introduced since the real world that is perceived through the thick panel of glass is refracted, but the graphics that are reflected by the front surface mirror are not. Thus, refraction would lead to slight registration errors between the refracted real world and the non-refracted virtual world if we determined the selectors as described above. Instead, we compute the corresponding refractors (the optically distorted rays) from the selectors and then find the closest point between the two refractors. If we do this, the resulting new landmark points would appear misaligned with their real counterparts by looking at the mirror. This is because the cross wires are aligned in a way that already incorporates refraction, since the adjusted surface points are perceived refracted by looking through the mirror. Due to the fact that this optical distortion is always present (not only during object registration), we apply the refracted model transform described in section 4.1.3, which also pre-distorts the registered landmark points. On the one hand, computing the refracted landmark points let them match spatially with the corresponding surface points. On the other hand, pre-distorting them afterwards matches them optically in addition. How to compute refractors is described in section 4.1.3.
Figure 5.17 shows that a sufficient object registration is possible with this method. We registered the outlining surface points of the printer (from two only slightly different perspectives for each point) and connected them with straight lines. Note that the reconstructed contour of the printer appears a little misaligned in the image, because the registered points were connected by straight lines, but the printer's edges are physically curved. However, instead of reconstructing the real object's surfaces like this, only a few landmark points are normally sampled and used to transform a pre-modeled virtual representation as described above. -The reconstruction example that is illustrated in figure 5.17 is only supposed to demonstrate the method’s applicability.
A similar method has been described by Fuhrman, Schmalstieg and Purgathofer [Fuh99b]. They use a tracked physical marker that has to be aligned with a set of displayed virtual mark-

ers to determine the camera parameters and to compensate optical distortion of head-mounted displays.

5.3.3.5 Remote Tools, distance and close Manipulation

The disadvantage of pointing and ray-casting for indirect interaction is that they cannot be used for directly defining spatial points. To support an immediate definition of points within the 3D free-space, we offer a remote usage of the same tools (i.e., the pen and the pad tools in our case) that are applied directly above the virtual workbench. As the direct tools (i.e., the tools directly used at the virtual workbench), the remote tools follow the motions of the input devices -but in contrast they are transformed by a constant offset to enable an operation behind the mirror. To allow an ergonomic usage of the remote tools, they can be frozen in their current position and orientation until the input devices are re-located to a more convenient posture. After unfreezing the remote tools, the new position and orientation offset is computed and used.

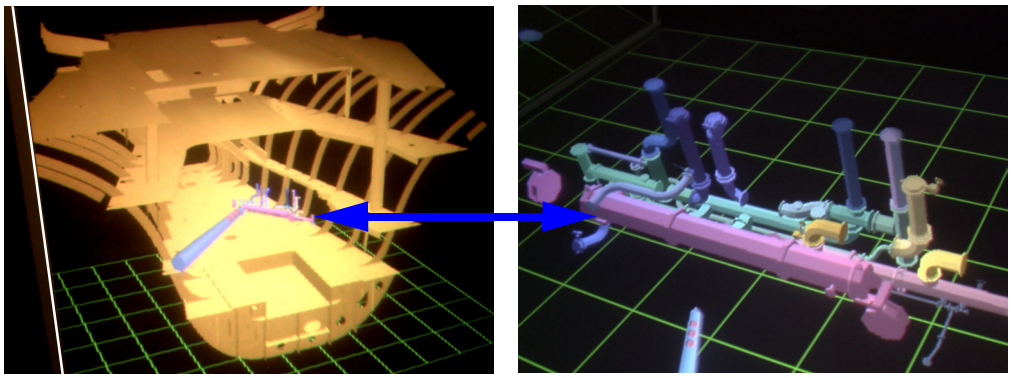


Figure 5.18: Distance manipulation with remote tools behind the mirror (left) and close manipulation above the virtual workbench with direct tools (right).

Sketching and drafting are examples for which spatial input tools are better suited than the ray or pointing techniques described in subsections 5.3.3.3 and 5.3.3.4.
While remote tools allow for distance manipulation, direct tools support close manipulation. Figure 5.18 shows a scenario that illustrates an example of an intuitive and smooth transition between distance and close manipulation: Using the remote pen, distance manipulation of ship-components (pipeline-clusters in figure 5.18) within a ship-section, which is visualized behind the mirror, is supported as long as the user looks at the mirror. If the user picks a component within the ship-section and then looks at the virtual workbench, the component is automatically transformed to fill out the entire space of the projection area at the virtual workbench. It is translated, rotated and scaled in a such way that a convenient close manipulation with the direct pen tool is supported. If, however, the user picks the component above the virtual workbench and then looks at the mirror, the component is automatically transformed back to the ship-section and is downscaled to fit into the ship-coordinate system.

5.3.4 Distortion Compensation and Correction

The following subsections will analyze sources of optical and non-optical distortion and present the implemented compensation methods.

5.3.4.1 Optical Distortion

Optical distortion is caused by the elements of an optical system. It does not affect the sharpness of a perceived image, but rather its geometry. The distortion can be corrected optically

(e.g., by applying additional optical elements that physically rescind the effect of other optical elements) or computationally (e.g., by pre-distorting generated images). While optical correction may result in heavy optics and non-ergonomic devices, computational correction methods might require high computational performance.
In Augmented Reality applications, optical distortion is critical, since it prevents precise registration of virtual and real environment.
The purpose of the optics used in HMDs, for instance, is to project two equally magnified images in front of the user's eyes, so that they fill out a wide field of view (FOV), and fall within the range of accommodation (focus). To achieve this, however, lenses are used in front of the miniature displays (or in front of mirrors, that reflect the displays within see-through HMDs). The lenses, as well as the curved display surfaces of the miniature screens may introduce optical distortion which is normally corrected computationally to avoid heavy optics which would result from optical approaches.
Since for HMDs, the applied optics form a centered (on-axis) optical systems, pre-computation methods can be used to efficiently correct geometrical aberrations during rendering.
Rolland and Hopkins [Rol93] describe a polygon warping technique as a possible correction method for HMDs. Since the optical distortion for HMDs is constant (because the applied optics is centered), a two-dimensional lookup table is pre-computed that maps projected vertices of the virtual objects' polygons to their pre-distorted location on the image plane. Note that this requires subdividing polygons that cover large areas on the image plane. Instead of pre-distorting the polygons of projected virtual objects, the projected image itself can be pre-distorted, as described by Watson and Hodges [Wat95], to achieve a higher rendering performance.
Correcting optical distortion is more complex for our mirror extension, since, in contrast to HMDs, the image plane that is reflected by the mirror is not necessarily centered (on-axis) with respect to the optical axes of the user. In fact, the alignment of the reflected image plane dynamically changes with respect to the moving viewer, while the image plane itself remains at a constant spatial position in the environment. There are mainly three sources of optical distortion in our case: geometric distortion caused by the projector, mirror flexion and refraction caused by the optical combiner.
To correct the geometric distortion that is caused by the projector which is integrated into the Virtual Table, we apply the projected image transform described in section 4.1.4. To sample grid points, we apply a device that is usually used to track pens on a white-board - the Mimio [Dun01]. The Mimio is a hybrid (ultrasonic and infrared) tracking system for planar surfaces which is more precise and less susceptible to distortion than our electro-magnetic tracking device. It provides a stable positional resolution of 0.3 mm with an update rate of 87 measurements/second. As illustrated in figure 5.19, its receiver has been attached to a corner of the Virtual Table (note the area where the Mimio cannot receive correct data from the sender, due to distortion. This area has been specified by the manufacturer).

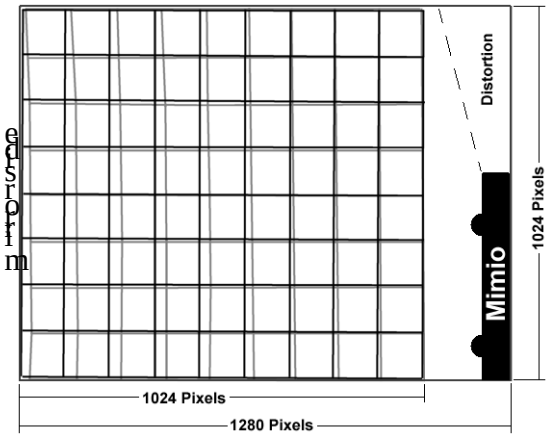


Figure 5.19: Projector calibration setup for Extended Virtual Table.

Since the maximal texture size that our rendering package supports is 1024 x 1024 pixels, the undistorted grid is rendered within the area (of this size) that adjoins to the mirror. We found that 10 x 9 sample points for an area of 40" x 40" on the projection plane is an appropriate grid resolution which avoids over-sampling but is sufficient enough to capture the distortion.
For flexible beam-splitters (e.g., large and thin ones) which are mounted in such a way, gravity causes a slight flexion of the mirror that affects their 1st order imaging properties (i.e., magnification and location of the image) and consequently causes a deformation of the reflected image. This represents another source of optical distortion.
As illustrated in figure 5.20, the viewpoint $\grave{e}$ observes the virtual object $\grave{p}$. Applying the reflected model-view transform (see section 4.1.2), $\grave{p}$ is projected to $\grave{p}'$. But because of physical flexion of the mirror, the observer perceives the reflection of $\grave{q}'$ instead of $\grave{p}'$. To overcome this kind of distortion, rendering methods for curved optics (see section 4.2) have to be applied instead of methods for planar optics. For the Extended Virtual Table setup a thick float-glass material has been selected to keep optical distortion caused by bending minimal. Thus, no additional method to pre-distort for mirror flexion has been applied during our experiments.

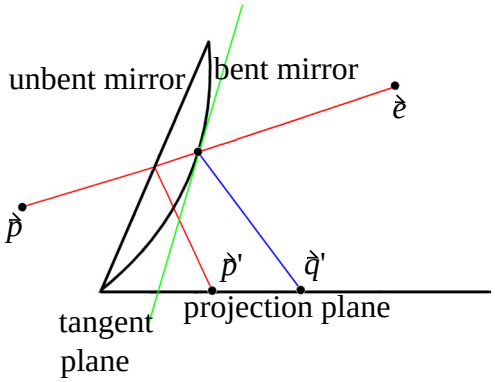


Figure 5.20: Optical distortion caused by mirror flexion.

On the one hand, a thick panel of glass stabilizes the mirror and consequently minimizes optical distortion caused by the mirror's flexion. On the other hand, however, it causes another optical distortion which results from refraction. To correct the optical distortion caused by

refraction, the refracted model transform (see section 4.1.3) has been applied. An evaluation of this approximation is presented in section 6.1.
Note that we correct optical distortion only while the user is working in the see-through mode (i.e., while looking through the mirror at an illuminated real environment). For exclusive VR applications, optical distortion is not corrected, even if the mirror is used as an extension.

5.3.4.2 Non-optical distortion

Accurate registration requires accurate tracking. In addition to the non-linear tracking-distortion, end-to-end system delay (time difference between the moment that the tracking system measures a position/orientation and the moment the system reflects this measurement in the displayed image) or lag causes a "swimming effect" (virtual objects appear to float around real objects).
However, since ideal tracking devices do not yet exist, we apply smoothing filters (sliding average windows) to filter high-frequent sub-bands (i.e., noise) from the tracking samples and prediction filters (Kalman filters [Azu95] for orientation information, and linear prediction for position information) to reduce the swimming effect.
Our tracking device, Ascension's Flock of Birds [Asc01], provides a static positional accuracy of 2.5 mm (by 0.75 mm positional resolution) and a static angular accuracy of 0.5° (by 0.1° angular resolution). The highest update rate (without system delay) is 100 measurements/second.

5.3.5 Discussion

We have presented an optical extension of table-like projection systems. Using a large optical combiner, we can enrich the scope of such display devices by allowing a viewing and interaction beyond the display’s boundaries. In addition, the extension enables such projection systems to support Augmented Reality tasks (projection-based AR). In contrast to immersive display technology, semi-immersive workbenches can easily be integrated into habitual work environments. Furthermore, our extension offers the combination of a virtual and a real workbench - representing a hybrid workplace.
If we compare our setup to head-mounted displays -as the traditional AR device- we can conclude that the optical characteristics of our approach can catch up with high-end HMDs (see section 6.6).
However, the Extended Virtual Table lacks in three major factors: *mobility, direct interaction with augmented real objects,* and *multi-user support*.
We believe that stable and precise long-range tracking will exist in the near future, enabling AR applications using HMDs to be highly mobile. Nevertheless, the intention of the Extended Virtual Table is to combine two table-like workplaces where the users focus on the workspace above the workbenches. For this, we neither require long-range tracking nor a high degree of mobility.
HMDs also offer direct interaction with augmented real objects that are in arm’s reach of the user. In case of the Extended Virtual Table, the mirror represents a physical barrier for the user's hands and the input devices and to a large degree prevents direct interaction with superimposed objects. We can either directly interact with real objects on the real workbench and with virtual objects on the virtual workbench, or indirectly interact with virtual objects above the real workbench through the mirror. Additional mechanical installation on the real workbench (such as a turntable) can be useful to remotely interact with real objects. Input devices that are better suited for indirect interaction (such as the Cubic Mouse, [Fro00]) can be used in addition to traditional input tools. Those devices can provide force feedback in upcoming versions which makes a direct interaction more convincing. Beside force feedback that is mechanically simulated by the input device, pseudo-haptics seems to be another interesting point to

consider for indirect interaction with the Extended Virtual Table. In pseudo-haptics, force feedback is simulated on a purely visual basis, rather than with mechanical aids (Lécuyer, Coquillart, Kheddar, Richard & Coiffet [Lec00]).
While HMDs provide individual image planes for each participant of a multiple user session, users of large projection systems have to share the same image plane. Thus, multiple user scenarios are difficult to realize with such technology. Since the Extended Virtual Table also applies a single large projection plane, it faces the same problem. Although some solutions exist that simultaneously support two users (e.g., Agrawala et al [Agr97]), they are not widely applied since they require special hardware.
The Extended Virtual Table's non-simultaneous (gaze-directed) viewing of the two projection spaces (in front of the mirror and behind it) can be either interpreted as a disadvantage - since the application of a second projector and another diffuse projection plane would support a simultaneous viewing (as it is the case for L-shaped workbenches) - or as an advantage - since no second projector is required. However, an additional opaque projection plane would make a see-through mode impossible. Instead, a front-projected or rear-projected semi-transparent (non-reflective), or a rear-projected holographic projection plane could be applied -but at the cost of image quality and the viewing range (compared to the much better optical characteristic of half-silvered mirrors).
We envision the possible application areas of the Extended Virtual Table to range from visualization of scientific data or simulation results in combination with real-world objects, over tele-cooperation to hybrid modeling and assembly.
In hybrid modeling and assembly applications, virtual mock-ups (VMUs) can be modeled above the virtual workbench and then be assembled to corresponding physical mock-ups (PMUs) that are located on the real workbench. Combining VMUs and PMUs to hybrid mock-ups (HMUs) allows for a more realistic early design review and possible early refinements within the conceptual product design phase.
The Extended Virtual Table can also be used to support a shared design review and modeling between several distributed parties within a semi-immersive tele-cooperation session. The telepresence factor is limited in most tele-cooperation systems that apply desktop screens, since communicating through a screen barely gives an impression of presence. A stereoscopic life-sized video-stream of a remote collaborator can be visualized in the mirror of the local user - giving her the impression that the collaborator is standing right beside her virtual workbench. Two or more participants could see each other (and each other's local design space) within their local mirrors and could bring together and discuss different virtual components that could be exchanged intuitively by passing them through their mirrors. In this way, they make use of an efficient, location-spanning form of teamwork.

5.4 The Transflective Board

This section presents another PBAR device (the *Transflective Board*) that utilizes a planar mirror [Bim01b]. It addresses the flexibility problem of the Extended Virtual Table and offers a sketch-based interaction interface. Our prototype represents a seamless combination of a whiteboard and an optical combiner -merging virtual objects with the surrounding real environment. With this, we strive for an efficient and problem specific application of Augmented Reality technology within the engineering domain.

5.4.1 Motivation

Being a variant of the Extended Virtual Table, the Transflective Board shares most of the xVR-related properties that have been discussed for its predecessor: It supports a combination of VR and AR by offering a technical and conceptual extension of traditional VR by means of

AR. Additionally, it supports a seamless integration into habitual work-environments by providing an interface between a virtual and a real work-place. As all PBAR configurations, it approaches to improve some of the technological, ergonomic and economic shortcomings that can be attributed to traditional AR devices.
In addition, the Transflective Board has been developed to further extend the scope of projection-based AR towards the engineering domain. It addresses the flexibility problems of the Extended Virtual Table (i.e., mobility[25] and direct-manipulative interaction) and offers a new, sketch-based interaction interface.

5.4.2 Increasing Flexibility

Our current experimental prototype has the same form-factor as a regular white board. It consists of a mobile Aluminium rack that holds a tiltable wooden frame. The actual white board is replaced by a half-silvered mirror (a 40" x 60" and 10 mm thick glass pane which has been laminated with a half-silvered mirror foil [3M01] that simultaneously transmits and reflects light. Thus, we refer to it as Transflective Board. Technically, the Transflective Board is used as an optical combiner that reflects stereoscopic 3D graphics off an arbitrary display surface. Using the optical combiner, these graphics are spatially merged with the surrounding real environment. Conceptually however, the Transflective Board can still be used like a white board - providing the same interaction behavior.
Two different stereo displays have been tested with the Transflective Board so far:

- A Barco BARON Virtual Table [Bar01] that applies rear-projection and active shuttering using CrystalEyes Shutter glasses [Ste01]. In this case, two stereo images are projected sequentially onto the projection surface and shuttering is synchronized via an infrared signal.This is illustrated in figure 5.21;
- A mobile two-screen front-projection system that applies passive shuttering using polarized light. In this case, four beamers project pre-filtered stereo images simultaneously onto the two projection planes. The images are separated via polarized glasses. This is illustrated in figure 5.22.

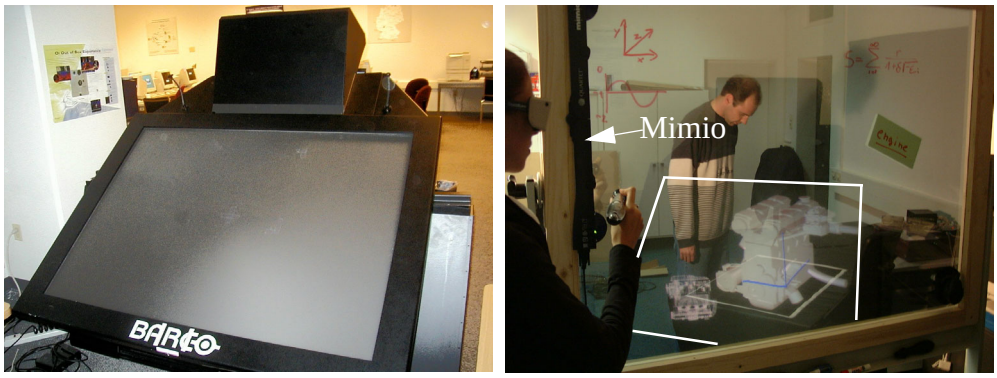


Figure 5.21: The Transflective Board: A combination of white board and optical combiner - merging virtual objects with the surrounding real environment. The projection device used for this scenario is a Barco BARON Virtual Table. The white lines outline the reflected projection plane.

25. Non-stationary - not in the sense of portable or wearable.

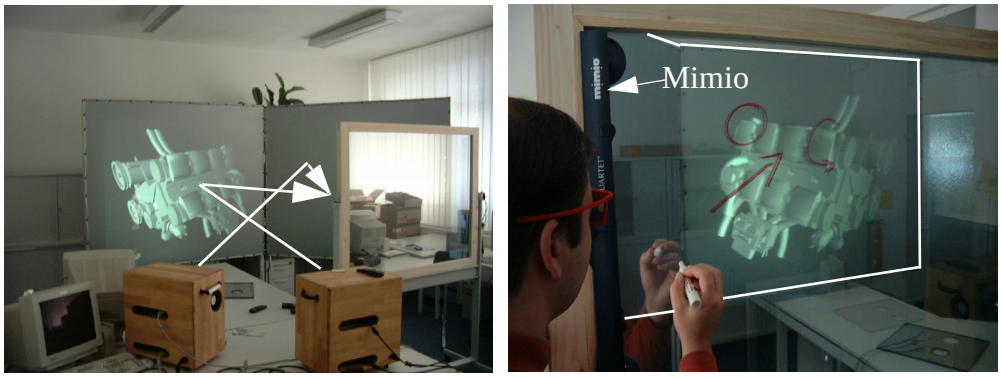
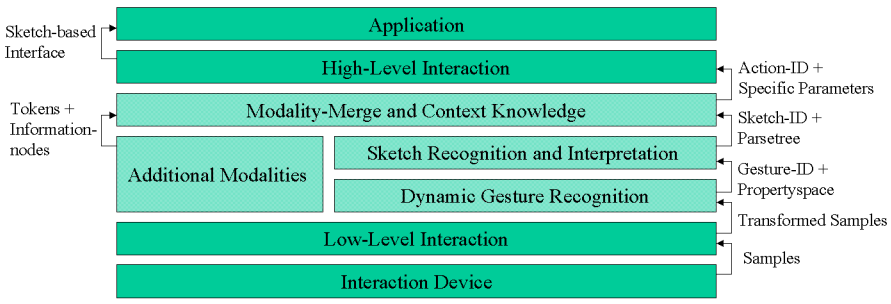


Figure 5.22: The projection device used for this scenario is a mobile two-screen system. The white arrows illustrate the flow of light. The white lines outline the reflected projection planes.

An electro-magnetic tracking device (Ascension's Flock of Birds [Asc01]) is used to support head-tracking and tracking of spatial input devices. In addition, we apply a tool that is usually used to track real markers and cleaning pads on a whiteboard - the Mimio [Dun01]. The Mimio is a hybrid (ultrasonic and infrared) 2D tracking system for planar surfaces which is more precise and less susceptible to distortion than our electro-magnetic tracking device. Note that the same device has also been applied for the projector calibration of the Extended Virtual Table. As illustrated in figures 5.21 right and 5.22 right, its receiver has been attached to a corner of the Transflective Board.
In principle, the same Augmented Reality techniques and tools (i.e., reflected model-view transform, optical and non-optical pre-distortion, registration and calibration, real-virtual object occlusion and collision detection, etc.) are supported with the Transflective Board as for the Extended Virtual Table. Beside the palette of interaction techniques discussed for the Extended Virtual Table, the Transflective Board provides a sketch-based interaction interface - adapting the interaction to the whiteboard-like output device.

5.4.3 Sketch-Based Interaction

To provide a flexible sketch-based user interface, we employ a multi-layered architecture which consists of eight hierarchically arranged layers (cf. figure 5.23) [Bim00c]. Each layer can interact with its direct upper or lower neighbor, whereby every layer can be deactivated, making the next activated layer a direct neighbor. This modularity offers applications the opportunity for individual utilization of the required functionality.



The core layers (emphasized in figure 5.23) contain the intelligent parts of the architecture, which are being widely implemented by applying methods of artificial intelligence (such as neural networks and fuzzy logic) [Bim99a].
Each layer (or single component within the layers) can be updated by alternative and improved versions, thus an adaptation of the architecture to an ongoing evolution of the components (e.g., caused by technological developments) can be supported.
The limited dependencies between layers and components also allow for distributed processing. In contrast to a centralized approach, a distributed modality processing, for instance, offers an extensive speed-up of the application, as well as the utilization of heterogeneous software and hardware. In our example, both applied modalities (speech and gestures) are independently analyzed on different processors, and are merged on a central node.
The experimental hardware environment of the Transflective Board represents the lowest level of the framework. The following *interaction devices* are utilized to support a sketch-based interaction: a 2D/3D-tracked whiteboard marker [Dun01] and the 3D-tracked Translucent Pad [Schm99].
Based on these devices, the *low-level interaction* layer implements several interaction techniques that support the sketching process within a constrained 2D space (i.e., on the Transflective Board or the Translucent Pad) or within the 3D-freespace. Examples of implemented techniques are 2D/3D drag-and-drop of strokes and sketches, and 2D/3D stroke-snapping constraints [Bim00c].
The *dynamic gesture recognition* layer implements several methods for stroke enhancement and multidimensional dynamic gesture recognition [Bim99a]. Single strokes that have been identified are parsed in a bottom-up manner, until an entire predefined sketch can be *recognized*. Each sketch is represented by its resulting parse-tree which will be *interpreted* to reconstruct the encapsulated geometric and context-sensitive information.
Speech is used in our setup as *secondary modality*, mainly to complement the gestural information. For this, we apply an off-the-shelf software package for continuous speech recognition (IBM's ViaVoice [Ibm01]). As for sketch recognition, a grammar has to be defined in advance to outline the recognizable speech language.
The *modality-merge and context knowledge* level merges the different information extracted from the single modalities (i.e., speech and gestures), as well as the provided context knowledge. The result of this layer triggers the action defined within the *high-level interaction* layer. The following sketch-based high-level interaction techniques are implemented:

- Geometry/object reconstruction from 2D [Enc99] or 3D [Bim99c] sketches;
- Object/Environment interaction [Bim99b];
- System control [Enc99];
- Text input via handwriting recognition [Blink95].

Beside other interaction methods (see section 5.3.3), these high-level sketch-based techniques can then be offered to the engineering *applications*, that sit on top of our framework.

5.4.4 Envisioned Engineering Applications

Today, maintenance and training are the major application fields for Augmented Reality from the industry's point of view [Arv01]. Augmented Reality will conquer the earlier phases of the product development process where VR is more heavily used. AR can compensate for the shortcomings of VR, such as fine and precise tactile and haptic feedback. Here, mobility is not a crucial issue. Thus, stationary set-ups that ensure high quality images and a more robust tracking than mobile ones will be preferred for these purposes.
We envision five application scenarios for projection-based AR in the engineering process:

- *Augmented design review:* VR systems are mainly used for design review. But the lack of reality especially concerning haptic and tactile feedback still limits the acceptance of VR as a decision base. This lack can be compensated by full functional rapid prototyping (RP) parts that give the user a real tactile and also aural feedback. To produce many RP parts with, for instance, different colors and materials is cost intensive. The combination of RP parts with virtual overlays that let these parts appear in different colors/materials is a promising 'best-of-two-worlds'-approach;
- *Hybrid assembling:* The feasibility evaluation of the assembling/disassembling processes is a common application of VR whereby the product exists only in a digital form. But products are frequently altered during their life cycle. In these situations, the products already exist physically and the need to assemble virtual models into physical mock-ups (PMU) arises. Figure 5.24 illustrates an example that could be extended towards a hybrid assembling scenario which allows for mounting a virtual engine into a real engine compartment via a direct manipulative interaction;
- *Hybrid modeling/sketching:* Although CAS (Computer Aided Styling) and VR systems have changed the styling and design process considerably, clay models still have their place in the design process. During a review of a physical model, requirements to change some of its parts/features appear. Today, these modification requirements are expressed verbally. An Augmented Reality setup would allow to virtually sketch the change onto/over the PMU;
- *Visual inspection of molded parts:* The development of a product comprises the design of tools to manufacture the product parts. Molded parts shrink when they cool down. Thus, the tools are not simply the inverse product geometry. When starting a new production line, the produced parts have to be compared with the intended geometry - the virtual parts.
- *Hybrid ergonomic design:* With virtual humans, ergonomic analysis can be carried out in virtual environments. After having built a virtual model, the manikin is used to evaluate the reachability of surrounding elements. In case of a poor ergonomics, the model has to be changed and the process is repeated. With an Augmented Reality setup the process can be inverted: e.g., a coarse physical mock-up of a cockpit is built and only afterwards virtual buttons are placed in the hybrid environment. This approach leads more directly to an ergonomic design.

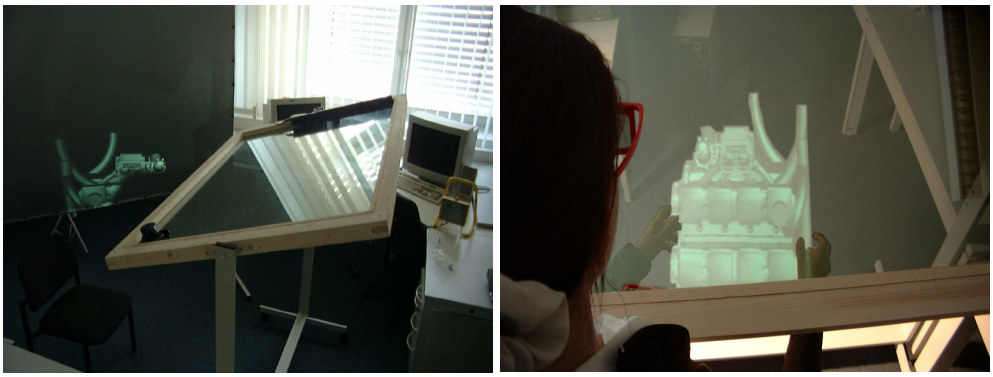


Figure 5.24: The transflective board can be used as a large reach-in system -supporting a direct-manipulative interaction with real and virtual objects. The graphics are projected onto the ceiling.

5.4.5 Discussion

Traditional Augmented Reality displays, such as head-mounted devices entail a number of technological and ergonomic shortcomings that prevent their usage in many of application

areas. With the objectives to overcome some of these shortcomings and to address an efficient and problem specific application of Augmented Reality technology within the engineering domain, we have presented the Transflective Board. Compared to head-attached AR displays, the application of spatial projection displays for Augmented Reality tasks provides an improved ergonomics, a theoretically unlimited field of view, a high and scalable resolution, and an easier eye accommodation [Ras98c]. Since the base technology that is modified for PBAR setups, as well as the consequential advantages are derived from the well established projection-based VR concept, a seamless combination of VR and AR (xVR) is also imaginable (further examples have been presented in [Enc00a, Enc00b, Bim01a]).
Beside these technological and conceptual advantages, a potential mobile AR setup (cf. figure 5.22) and the possibility of a direct manipulative interaction with the augmented real environment (cf. figure 5.24) makes Augmented Reality attractive for the engineering domain.
The integration of a sketch-based interaction interface has shown that a broad palette of high-level interaction techniques, such as object creation, object interaction, freeform modeling, text input, and environment control can be offered on a human-centered and intuitive basis. These techniques were integrated into existing domain-specific applications (also engineering applications) to extend their interaction functionality (while maintaining their existing user interfaces) and to evaluate the techniques' applicability. In contrast to non-immersive desktop approaches, immersive or semi-immersive three-dimensional environments offer a less constrained sketching (mainly due to the possibility of one- or two-handed 3D interaction, 3D navigation and stereoscopic visual perception) [Sto00]. Whiteboard-like and workbench-like systems together with pen and pad combinations turned out to be well suited tools, since they support both a constrained 2D, as well as an unconstrained 3D sketching process on an intuitive basis, while representing common and well known everyday items, such as drafting boards, sketch pads, clip boards, and real pens. In the application area of 3D modelling, sketching within free space is a consequent step towards supporting the user in her behavioral and perceptual possibilities.
However, several shortcomings can be related to Transflective Board. Self-reflection of the user in some situations, for instance, is one of the shortcomings of the Transflective Board. Our current approach to address this problem is to integrate the Transflective Board into engineering-related application scenarios so that these reflections are minimized or avoided.

5.5 Virtual Showcases

Virtual Showcases [Bim01c, Bim01d] are an alternative medium that allow to superimpose real exhibits with any kind of graphical information, or to display exclusively virtual exhibits. As for the other PBAR devices, the Virtual Showcases' general aim is to detach the display technology (such as head-worn devices) from the user and to integrate it stronger into our habitual living environments.
In this section, we present the idea of Virtual Showcases as yet another PBAR configuration. The presented prototypes consist of planar multi-mirror setups or curved mirror configurations and are used in combination with traditional display devices.

5.5.1 Motivation

Intuitive access to information in habitual environments is a grand challenge for information technology. An important question is how established and well functioning everyday environments can be enhanced rather than replaced by virtual environments. Augmented reality technology has a lot of potential in this respect since it allows the augmentation of real world environments with computer generated imagery. Today, most Augmented Reality systems use

see-through head mounted displays, which share most of the disadvantages of standard head mounted displays.
In this section, we introduce Virtual Showcases as new Augmented Reality display systems. The Virtual Showcase has the same form factor as a real showcase making it compatible with traditional museum displays. Real scientific and cultural artefacts are placed inside the Virtual Showcase allowing their three-dimensional graphical augmentation. Inside the Virtual Showcase virtual representations and real artefacts share the same space providing new ways of merging and exploring real and virtual content. The virtual part of the showcase can react in various ways to a visitor enabling intuitive interaction with the displayed content. These interactive showcases represent a step towards ambient intelligent landscapes, where the computer acts as an intelligent server in the background and visitors can focus on exploring the exhibited content rather than on operating computers.
In contrast to the PBAR configurations described above, Virtual Showcases offer the possibility to simultaneously support multiple viewers and to provide a seamless surround view on the augmented real environment located within the showcase.

5.5.2 Physical Arrangements

A Virtual Showcase consists of two main parts (cf. figure 5.25): a convex assembly of half-silvered mirrors (1) and a graphics display (2). So far, we have built Virtual Showcases with two different mirror configurations. Our first prototype consists of four half-silvered mirrors assembled as a truncated pyramid (cf. figure 5.25-left). Our second prototype uses a single mirror sheet to form a truncated cone (cf. figure 5.25-right). These mirror assemblies are placed on top of a projection screen - a Barco BARON Virtual Table (2) [Bar01] in our current setups. Users can see real objects inside the mirror assembly through the half-silvered mirrors merged with the graphics displayed on the projection screen. The showcases' contents are illuminated with a controllable light source (3) while view-dependent stereoscopic graphics is presented to the observer(s). For our current prototypes, stereo separation and graphics synchronization are achieved with active shutter glasses (5) [Ste01] and infra-red emitters (4), and head-tracking is realized with an electro-magnetic tracking device (6) [Asc01]. While our first, pyramid-shaped prototype simultaneously supports up to four viewers, our second, cone shaped prototype provides a seamless surround view onto the displayed artifact for less viewers.

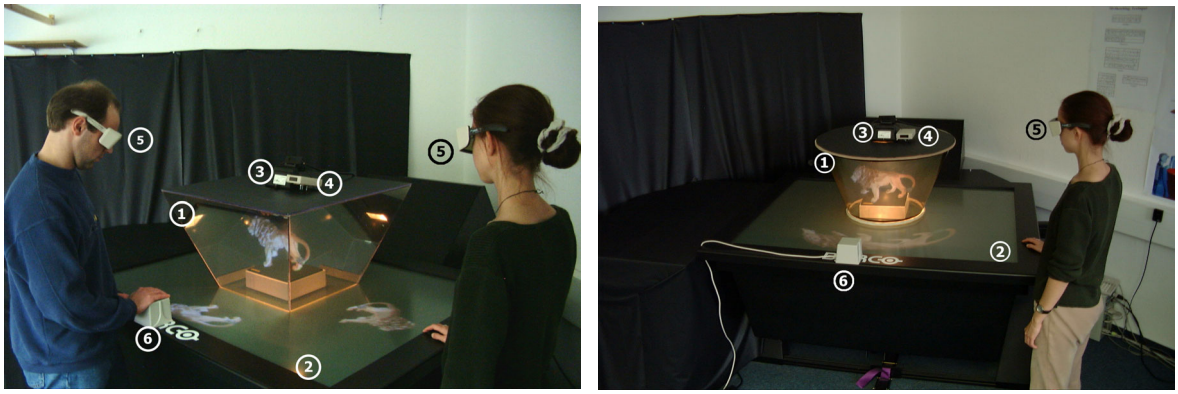


Figure 5.25: The Virtual Showcase prototypes. A Virtual Showcase built from planar sections (left), and a curved Virtual Showcase (right).

Depending on the mirror geometry (e.g., planar or curved) the reflection of the object space maps differently into the image. While the image space of planar mirrors is an affine map of the object space, the image space of curved mirrors is curvilinearly transformed. In case of

half-silvered mirrors, the image space unites the reflected image of the object space in front of the mirror and the transmitted image of the real environment behind the mirror (i.e., within the Virtual Showcase).

5.5.3 Virtual Showcases built from Planar Sections: Supporting multiple Viewers

We want to classify non-autostereoscopic multi-viewer approaches into three general categories: *private screens*, *frame interleaving*, and *screen partitioning*.

For *private screen* approaches, each viewer is assigned to an individual screen - presenting an individual frame-buffer content. Head-mounted displays are the classical private screen devices. However, projection-based private screen solutions have also been proposed: UNC's two-user Protein Interactive Theatre [Arth98] assigns two righted projection planes to individual observers. Thus, the two users' viewing-frustum cross each other while looking at the displays.

Stanford's two-user Responsive Workbench [Agr97] is an example for a *frame interleaving* approach. The different images that are rendered into separate frame-buffers are presented time sequential with the same display to individual viewers. Shutter technology allows an appropriate image separation.

We want to refer to the third and most recent category as *screen partitioning*. The different images that are rendered into a single frame-buffer are displayed on different portions of the same screen. For multiple observers, the image separation is achieved by using view-blocking elements. The IllusionHole setup [Kit01] applies a simple canopy that allows to perceive the individual image through a small hole, whereby the other images are covered by the canopy.

Virtual Showcases are screen partitioning systems that use multi-mirror or curved mirror configurations to achieve an image separation:

By assigning each viewer to an individual mirror, the pyramid-like Virtual Showcase prototype can support up to four observers simultaneously (cf. figure 5.26). The rendering techniques used for these multi-mirror setups are described in section 4.1.5.

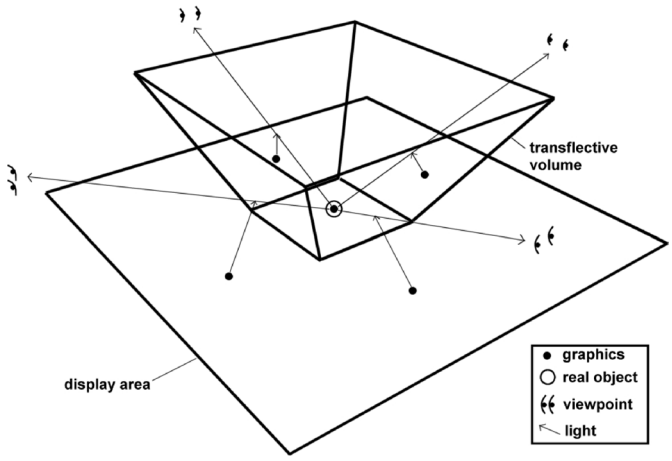


Figure 5.26: Serving four viewers simultaneously - conceptual sketch.

Since convex mirror assemblies unequivocally tessellate the object space into mirror individual reflection zones which do not intersect or overlap, a single object that is displayed within the object space appears exactly once within the image space. Consequently, a definite one-to-one mapping between the object space and the image space is provided by convex mirror assemblies.

Observed from a known viewpoint, the different images optically merge into a single consistent image space by reflecting the projection plane, whereby this image space visually equals the image of the untransformed image space geometry. This is demonstrated in figure 5.27.

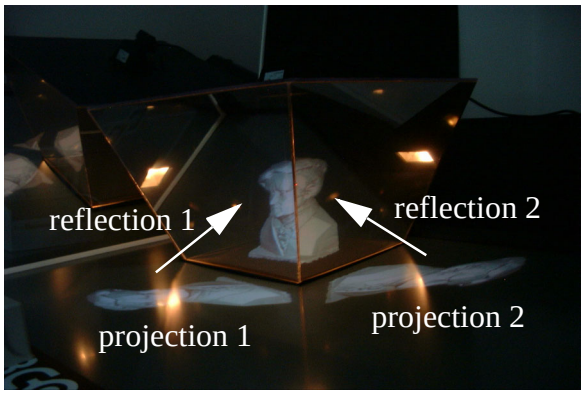


Figure 5.27: Different reflections are optically merged into a single consistent image space.

Figures 5.28-left and 5.28-right show two individual views onto the same image space (seen from different perspectives). For instance, these views can be seen by a single viewer while moving around the showcase, or by two individual viewers, while looking at different mirrors simultaneously.

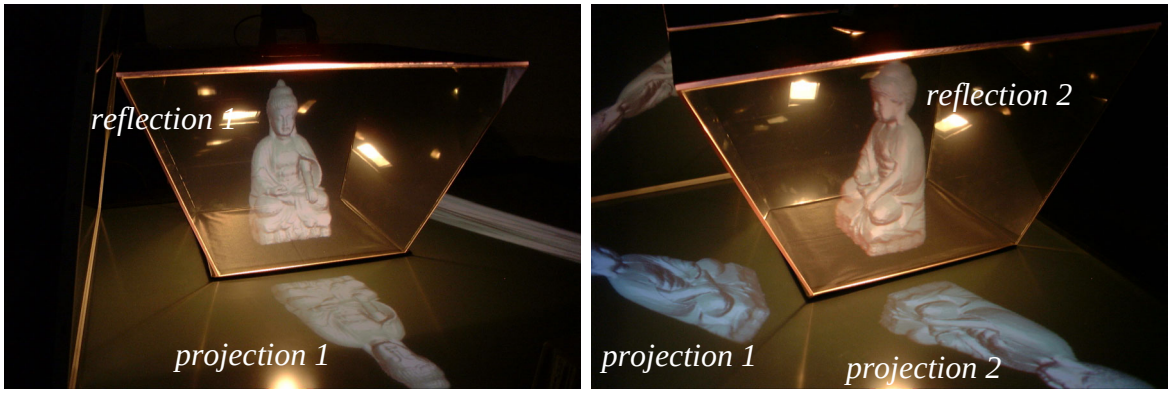


Figure 5.28: Two individual views onto the same image space.

Figure 5.29 illustrates another application example of Virtual Showcases. Here a model of a car can be observed from different viewpoints. The application rotates the model on a virtual turntable and allows the users to interactively modify the varnishing parameters of the car's body. Multiple users can either interact with their individual contents, or they can perceive the same content. Note that a static mirror-viewer assignment is not required - even individual mirror sections can be dynamically assigned to moving viewers. In case multiple viewers look at the same mirror, an average viewpoint can be computed (this results in slight perspective distortions).

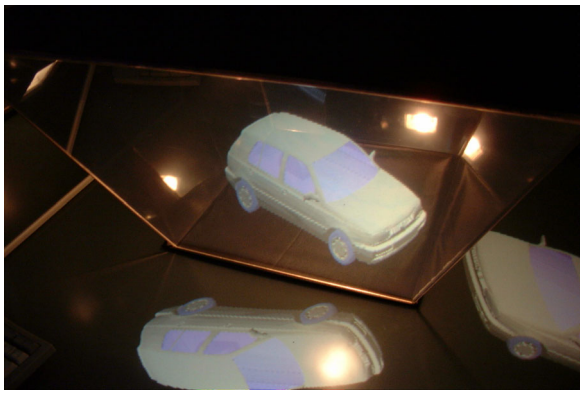


Figure 5.29: Virtual Showcase used to display a model of a car.

While figures 5.27-5.29 show exclusively virtual exhibits, figures 5.30 and 5.31 show an example of a mixed (real/virtual) exhibit, displayed within a Virtual Showcase.

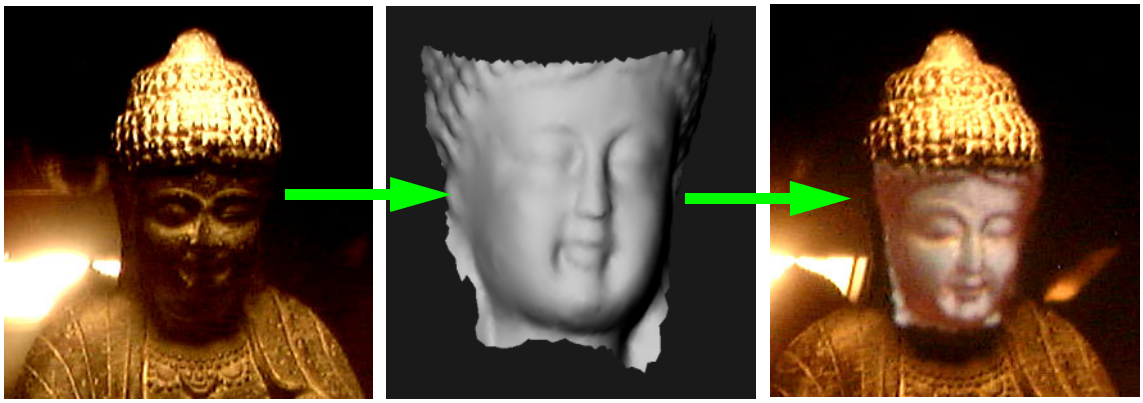


Figure 5.30: The Buddha's face has been scanned and superimposed onto the real statue.

The surface of the real Buddha statue in figure 5.31 was scanned three-dimensionally and the virtual model has then been partially reflected back onto the real statue to demonstrate the precise superimposition and registration of the two environments (cf. figure 5.30). Figure 5.31 illustrates the whole scenario with additional multi-media information and a supplementary textured face.

Figure 5.31: A real Buddha statue complemented with geometric and other multimedia information.

To provide a single consistent image space (consistent, regardless of the viewpoint), the mirror planes have to be registered very precisely. Slight misregistrations optically result in gaps between the reflections, and therefore in multiple inconsistent image spaces. This problem is comparable to mis-registered multi-plane projection devices or tiled displays, although for mirror planes the optical aberrations are view-dependent.

5.5.4 Convexly Curved Virtual Showcases: Providing a seamless Surround View

Building Virtual Showcases from one single sheet, instead of using multiple planar sections reduces the calibration problem to a single registration step and consequently decreases the error sources. In addition, the edges of adjacent mirror sections (which can be annoying in some applications) disappear. Consequently, it provides a seamless surround view onto the displayed artefact. However, using curved mirrors requires a curvilinear transformation of the object space. The rendering techniques used for convexly curved mirror configurations are described in section 4.2.1.
An example of a hybrid exhibit (a virtual hand places a virtual cartridge into a real printer) is demonstrated in figure 5.32.

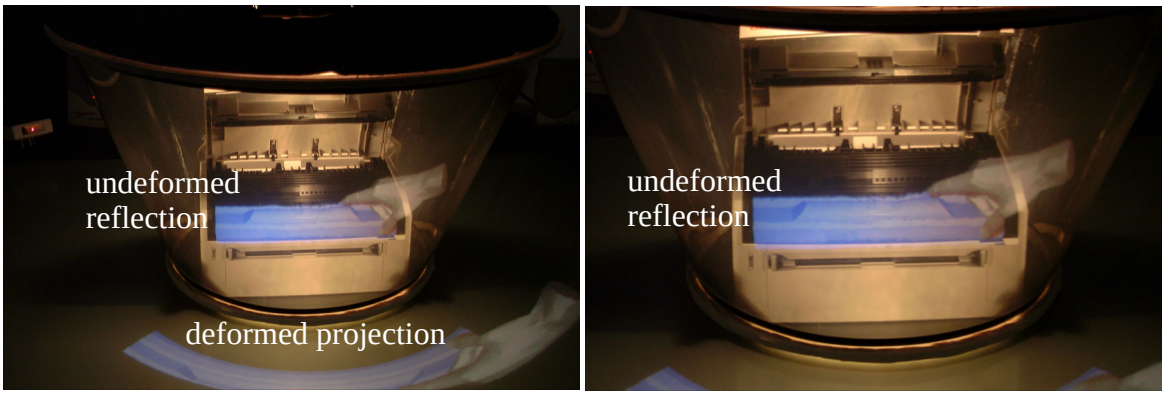


Figure 5.32: The curved projection is reflected into a straight reflection.

Figure 5.33 illustrates an exclusively virtual content observed from different viewpoints.

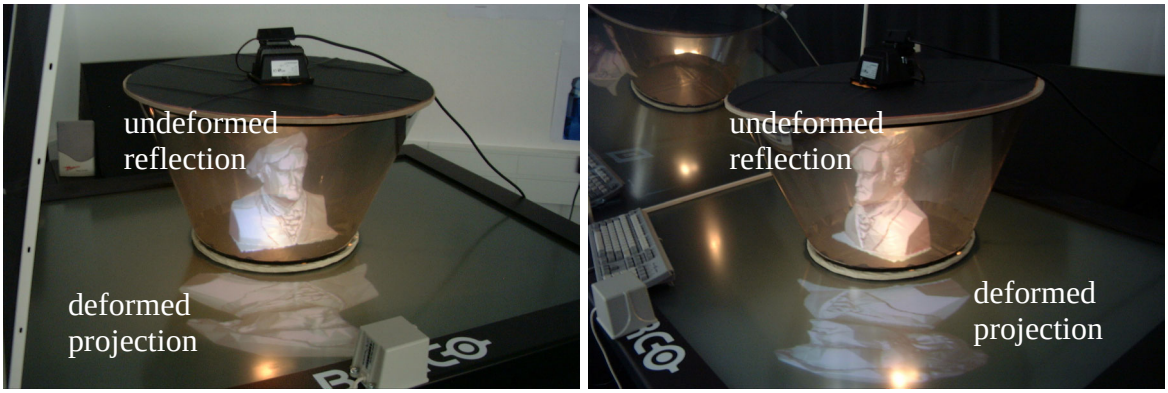


Figure 5.33: A virtual Wagner bust observed from different viewpoints.

If the multi-pass rendering method described in section 4.2.2 is applied, the first-pass rendering method can be exchanged. Instead of using a geometric renderer, other techniques (such as image-based and non-photo-realistic rendering, interactive ray-tracing, volume rendering, etc.) can be employed to generate the picture of the image space. Especially rendering techniques

that generate realistic images of complex scenes at interactive rates are of particular interest for Virtual Showcases. Beside an ordinary geometric renderer that has been applied to generate the images shown above, image 5.34 shows the application of a volumetric renderer [Eck98]. For the image displayed in figures 5.25 and 5.35, however, a progressive point-based (splatting) renderer [Rus00] has been chosen.

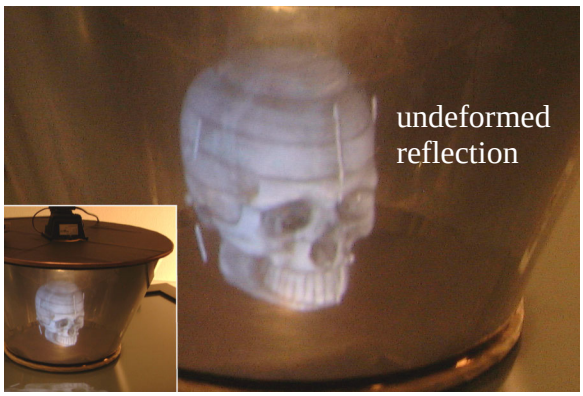


Figure 5.34: A volumetric renderer generates the image during the first rendering pass.

Figure 5.35: A progressive point-based renderer generates the image during the first rendering pass.

Note that since convex mirrors map a larger portion of the object space (i.e., of the display) into a smaller portion within the image space, a high density of pixels can be compressed into a small reflection (cf. figure 5.35). Consequently, the spatial resolution within the reflection is regional higher than the spatial resolution of the display device.

5.5.5 Discussion

We have introduced a new Augmented Reality display device - the Virtual Showcase. Virtual Showcases allow the superposition of real exhibits with arbitrary graphical information. Solely virtual exhibits may also be displayed. Virtual Showcases are simple to build and easy to combine with existing display technology.

The rendering techniques implemented for planar and curved optics (see chapter 4) are flexible enough to be smoothly integrated into existing software frameworks, and they are general enough to support different hardware setups (i.e., projection devices and mirror configurations). Additionally, they take as much advantage of hardware-implemented rendering pipelines as currently possible.

While the rendering passes and per-primitive transformations can be completely executed by graphics accelerators of today's graphics adapters, intermediate per-vertex transformations are not supported by prevalent rendering pipelines (such as the one implemented in OpenGL). Consequently they cannot benefit from current graphics acceleration hardware. However, next generation graphics engines, e.g., [Lin01, Nvid01a], support programmable per-vertex operations, which will allow hardware acceleration of the required per-vertex transformations.
In combination with selective refinement methods (see section 4.3.1) and other acceleration schemes (e.g., progressive rendering, parallel processing and efficient image transmission -see sections 4.3.2 and 4.3.3), a high-quality image generation can be achieved at interactive rates. In addition, image-based methods allow the generation of realistic images of real objects, which is an important issue for the presentation of virtual exhibits.
Our current prototype uses a dimmable light source for illuminating real artifacts placed inside the showcase. This illumination causes inter-reflections, which become visible from the outside in some cases. This problem can be solved by applying up-side-down configurations of Virtual Showcases (e.g., using a ceiling projection). Additionally, this prevents the observers from directly seeing the projection plane. Furthermore, a video-projector based illumination instead of a simple light source, similar to [Ras98c, Ras99], allows controlled illumination on a per-pixel basis.
Recent advances in hardware image-composition technology [Stol01] will efficiently support a simultaneous rendering for multiple viewers on cost-effective rendering platforms (i.e., PCs) by providing a real-time composition and blending of images that are generated in parallel.
A non-planar projection geometry that adapts better to the mirror geometry would be more efficient for Virtual Showcase setups. Curved display devices can be used to project the deformed object space and require less room than the planar displays, used for our prototypes (see section 4.4 for rendering techniques that support non-planar projection surfaces).
In addition, active shuttering using glasses and wired tracking devices clearly handicap the users. Here, optical tracking approaches and glass-free stereo-image separation (i.e., auto-stereoscopic) technology promise further improvements.
Interaction techniques and input devices for Virtual Showcases need to be developed that are unobtrusive and work well for novice users. Since direct interaction is physically restricted by the mirror surfaces, indirect techniques and the use of passive real world props seem more appropriate. These interface issues are the most challenging problems for the integration of Virtual Showcase technology into museums and other everyday environments.
Overall, we believe that Virtual Showcases represent an interesting approach for specific spatial Augmented Reality applications. While general purpose devices, such as HMDs, certainly have their distinct areas of use, it may be a good idea to have a closer look at alternative display technologies that might make Augmented Reality more attractive to end users.

5.6 Summary

We have introduced several PBAR proof-of-concept prototypes to give evidence for the feasibility of the rendering techniques, introduced in chapter 4. In addition, we have adapted several VR/AR specific techniques for interaction, real object registration, real-virtual object occlusion and collision detection, and optical/non-optical pre-distortion. The implemented demonstrators were focused on the three main application areas: engineering, scientific visualization, and cultural heritage. Each of the single devices has been discussed in detail within the corresponding section of this chapter. The following issues have been addressed:

- Diminution of the window violation problem linked to semi-immersive projection screens (e.g., the Reflective Pad);

- Overcoming the occlusion problem linked to rear-projection screens (e.g., the Transflective Pad);
- Combination of VR and AR (i.e., extended VR/xVR);
- Seamless integration of xVR into everyday workplaces (e.g., the Extended Virtual Table);
- Flexible application of PBAR configurations (e.g., the Transflective Board);
- Support of multi-user viewing and seamless surround view (e.g., Virtual Showcases).

Compared to the related AR display technology described in section 3.2, the following observations can be made:

- The field of view can be increased constantly or interactively by using large static or small dynamic (hand-held) mirror beam-splitters;
- Images can be displayed with a high resolution (i.e., higher than with head-attached displays). This can be attributed to the application of high-resolution projectors or multi-projector setups. In addition, convexly curved mirrors compress a high density of pixels into a small reflection area and consequently provide a high spatial resolution within this area;
- The resolution is not restricted by the optics (in contrast to the holographic film utilized by transparent projection screens);
- Because of the optical see-through approach of our PBAR concept, the real environment is perceived in the resolution of the human eye and not in the resolution of a video capture device (e.g., a video camera used for video see-through AR);
- The see-through metaphor is supported, rather than remote viewing;
- Some PBAR setups do support a direct interaction (e.g., Transflective Pad and Transflective Board) while others only support a remote or indirect interaction (e.g., Extended Virtual Tables and Virtual Showcases);
- Pseudo real images can be generated by convex or planar mirrors. They support a direct interaction;
- Light-weight shutter-glasses provide an improved ergonomic behavior;
- The fixed focal length problem is improved. This can be attributed to better spatial alignment of the images within the real environment;
- As for other spatial displays, less incidence of discomfort due to simulator sickness is given. This can be attributed to the spatially aligned image planes (compared to head-attached image planes in case of HMDs) [Patr00];
- No shadow-casting of physical objects or of interacting users is caused. This can be attributed to the application of rear-projection (in contrast to Spatially AR or head-mounted projective displays, where front-projection is used);
- The appearance of virtual objects is not restricted to the real environment’s geometry (in contrast to Spatially AR);
- Multi-user and seamless surround view scenarios are possible;
- The viewing area is less restricted by the optics (in contrast to the holographic film utilized by transparent projection screens);
- A flexible alignment of projector(s), projection plane(s) and mirror beam-splitter(s) is provided (in contrast to transparent projection screens);
- A higher see-through quality is provided by the half-silvered mirror optics (in contrast to the holographic film utilized by transparent projection screens).
- The applied optical see-through elements cannot present mutual occlusion of real and virtual environments. Due to the optical characteristics of half-silvered mirrors, virtual objects appear to be semi-transparent rather than opaque. Consequently, bright real surfaces (which radiate real environmental light) overlaid with dark virtual objects (which generate

low-luminous reflections) optically emphasize the real environment and let the virtual objects disappear.

- PBAR displays are easier to calibrate than HMDs. HMDs can have up to 12 degrees-of-freedeom while PBAR displays usually have between 3 and 6 degrees-of-freedeom. In addition HMDs have to be re-calibrated for different users and different sessions. Consequently, PBAR displays can provide a more precise registration between real and virtual environments while requiering a small ammount of user and session independent calibraion effort.

Addressing the last point, we can refer to Kiyokawa et al [Kiy00] who present ELMO -an optical see-through BOOM-like display for mutual occlusion. It also applies a half-silvered mirror as an optical combiner and a semi-transparent LCD panel to selectively block the incoming light on a per-pixel basis. Beside the shortcomings of most stereoscopic head-attached displays (see sections 3.1.2.1 and 3.2.2) ELMO faces additional problems that are linked to the LCD panel: light attenuation caused by the LCD panel, and low response time and resolution of the LCD panel. However, as the first functioning system of its kind, it effectively addresses the mutual occlusion problem of optical see-through displays. The general idea might represent a possible future improvement for our PBAR configurations. Instead of using light-blocking elements, however, we can also envision to selectively control the light on a per-pixel basis to solve the mutual occlusion problem. Thereby, we can replace the diffuse light bulbs that are used to illuminate the real environment with a single or multiple projector(s) (technically similar, but conceptually different as described in [Ras01]). The rendering pipeline that has been developed for PBAR configurations can be extended to incorporate a projector-driven static or view-dependent illumination that generates shadows at those areas on the real objects' surfaces where graphics is overlaid [Bim02a].
Obviously, the interaction potentials of the different PBAR prototypes is a criterion for classification: While some systems represent interaction tools by themselves (e.g., the Reflective Pad and the Transflective Pad), others are rather used as passive output devices that support interaction through additional devices. Some prototypes support a convenient direct manipulative interaction with the augmented real environment (e.g., the Transflective Pad and the Transflective Board), and others provide only an indirect manipulative or remote interaction (e.g., the Extended Virtual Table). Still others offer a simultaneous interaction of multiple users (e.g., Virtual Showcases). Therefore, we can conclude that not only the rendering techniques, but also the interaction methods which are applicable in combination with a specific PBAR configuration are strongly influenced by the utilized optics. This, in turn, drives the applicability and the application area of a PBAR configuration.

6 Evaluation

This chapter presents several non-coherent evaluations of selected techniques and proof-of-concept prototypes to give an impression of their effectiveness.
Section 6.1 compares the precision of our analytical refracted model transform approximation with the precise numerical method. Both methods have been introduced in section 4.1.3. In section 6.2, we give an overview of the computational cost and the order-of-growth of the rendering and transformation techniques presented in chapter 4. While we present performance measurements of the curve-optics techniques in section 6.3, we provide experimental results of the selective refinement and progressive rendering accelerations schemes in section 6.4. A parallel processing case-study is discussed, and our image coding approaches are evaluated in section 6.5. Finally, section 6.6 gives an impression of the optical characteristics of a selected proof-of-concept prototype -the Extended Virtual Table that has been discussed in section 5.3. We introduce a general method that allows for comparing field-of-view related characteristics of heterogeneous Augmented Reality displays that apply possible off-axis image planes.

6.1 Precision of Refracted Model Transform

In this section, we want to compare the effectiveness of our final analytical refracted model transform approximation with the described precise numerical method (see 4.1.3).
The PBAR setup used for the following experiments is the Extended Virtual Table (see section 5.3), since the refraction caused by the thick glass panel used for this configuration is high enough to cause optical distortions. To compare both methods, we refracted a dense set of 3D vertices that covered the entire working volume behind the mirror over time (i.e., from varying eye positions) with both -the approximation and the precise numerical method. The results are shown in table 6.1. The minimization procedure of the numerical method was executed with a threshold of $\varepsilon = 0{,}01\,\text{mm}$.
The spatial distance between the analytically refracted points and their corresponding numerically refracted counterparts serves as error function. The results are shown in table 6.2.

Displacement caused by refraction (mm)	Minimal	Maximal	Average
Numerical Method	3.75	10.34	6.08
Analytical Approximation	3.53	9.78	5.95

Table 6.1: Comparison between numerical refraction and analytical approximation.

	Minimal	Maximal	Average
Deviation (mm)	0.03	1.38	0.19

Table 6.2: Average deviation between numerical method and analytical approximation.

Note that the average deviation between precise numerical method and analytical approximation is far below the average positional accuracy of the applied electro-magnetic tracking device (Ascension's Flock of Birds [Asc01]), as described in section 5.3.4.2. Thus, a higher distortion is caused by the inaccurate head-tracker than by applying the approximation to cor-

rect refraction misalignments. However, if refraction is not handled at all, the resulting optical distortion is higher than the one caused by tracking-errors.
Note also, that the presented approximation is only correct for plane parallel plates. If the optics is bent, a refraction method for curved lenses has to be applied (e.g., see section 4.2.2.4).
The transformation time of the refracted model transform increases with the number of scene vertices (see table 6.3). However, the analytical approximation requires only a small fraction of the transformation time that is required by the numerical minimization.

6.2 Analysis of Computational Cost and Order-Of-Growth

In the following section, we want to give an overview of computational cost and order-of-growth of the techniques presented in chapter 4 (see table 6.3). Note that the analysis does not consider the acceleration schemes that are introduced in section 4.3. These schemes approach computational cost savings of the image-based techniques. However, this is not reflected in table 6.3, but will be discussed in detail within section 6.3.
For the subsequent analysis, we define the following:

- Let V_s be the number of scene vertices;
- Let V_{ts} be the number of vertices of a tessellated scene, whereby $V_{ts} \gg V_s$;
- Let V_i be the number of image grid vertices, whereby $V_i \ll V_s$;
- Let V_p be the number of predistorted projector grid vertices;
- Let S be the complexity of scene (if geometric rendering is used: $S = V_s$ or $S = V_{ts}$);
- Let I be the image size (i.e., the texture size);
- Let E be the number of optical elements (i.e., planar sections and curved elements);
- Let P be the number of projectors;
- Let VP be the number of view-points;
- Let G_u be the undistorted projector grid;
- Let G_P be the predistorted projector grid consisting of V_p vertices;
- Let G_i be the image grid consisting of V_i vertices;
- Let S_d be the display surface model (i.e., a geometric approximation of the display surface);
- Let $C_r(S)$ be the cost to render the scene S (i.e., to send S through entire the rendering pipeline);
- Let C_{rf} be the cost to apply the refraction computations to one vertex;
- Let C_{rl} be the cost to apply the reflection computations to one vertex;
- Let $C_{f2t}(I)$ be the cost to transfer the frame-buffer content of size I to the texture memory;
- Let C_s be the cost to compute the shading values for one vertex;
- Let C_t be the cost to compute the texture coordinates for one vertex;
- Let $C_{cs}(G)$ be the cost to find the enclosing grid cell within the grid G for one vertex;
- Let C_{li} be the cost to compute a single linear interpolation within a grid cell for one vertex;

- Let $C_{pt}(S_d)$ be the cost for the projective texture mapping onto the display surface model S_d;
- Let C_{fl} be the cost to transform an image-grid vertex with respect to a reflective or refractive optical element (i.e., C_{fl} can either be a summation of C_{rf} and C_t for refraction computations, or C_{rl} for reflection computations);
- Let $C_p(S_d)$ be the cost to project an image-grid vertex onto the display surface S_d (i.e., to perform appropriate intersection computations).

Note that the computational cost of optional rendering or transformation components that are assumed within table 6.3 can be set to zero in case they are not applied.

Technique	Computational Cost	Order of Growth
Reflected View Transform	0	No additional computational cost (we do not consider the scene complexity).
Reflected Model-View Transform	0	No additional computational cost (we do not consider the scene complexity).
Refracted Model-View Transform	0 for affine transformations, or $V_{ts} \cdot C_{rf}$ for curvilinear transformations	**Order of Growth:** For curvilinear transformations: $O(V_{ts})$
Projected Image Transform	$C_r(S) + C_{f2t}(I) + C_r(G_p)$	two rendering passes + transfer of framebuffer content to texture memory (+ texture filtering) **Order of Growth:** $O(I) + O(V_p)$ (we do not consider the scene complexity)

Table 6.3: Analysis of computational cost and order-of-growth.

Technique	Computational Cost	Order of Growth
Convex Multi-Section Optics, Single Viewpoint	$E(V_{ts} \cdot C_{rf}+0+ C_r(V_{ts}))+C_{f2t}(I)+ C_r(G_p)$	**Assumption:** All components (refraction, reflection and projector pre-distortion) are applied, a single projector is utilized. *For each optical element:* refract scene + reflect scene + render scene, generate texture and render predistorted projector grid (+ texture filtering) **Order of Growth:** $O(E \cdot V_{ts})+O(I)+O(V_p)$ (if refraction transformation and projector pre-distortion are applied), $O(E)$ (if refraction transformation and projector pre-distortion are not applied)
Geometry-Based Reflected Model-View Transform w. per-vertex shading	$V_{ts} \cdot C_{rl}+V_{ts} \cdot C_s$	**Order of Growth:** $O(V_{ts})$
Image Generation	$C_r(S)+C_{f2t}(I)$	1st rendering pass + transfer of framebuffer content to texture memory **Order of Growth:** $O(I)$ (we do not consider the scene complexity)
Geometry-Based Reflected Model-View Transform w. texture generation	$C_r(V_s)+C_{f2t}(I)+ V_{ts} \cdot C_{rl}$	texture generation = image generation + transfer of framebuffer content to texture memory **Order of Growth:** $O(I)+O(V_{ts})$
Image-Based Reflected Model-View Transform	$V_i \cdot C_{rl}$	**Order of Growth:** $O(V_i)$
Refracted Image Transform	$V_i \cdot C_{rf}+V_i \cdot C_t$	refraction + computation of texture coordinates **Order of Growth:** $O(V_i)$
Implicit Projected Image Transform	$V_i \cdot C_{cs}(G_u)+ V_i \cdot C_{li}$	find enclosing grid cell + linear interpolate within gird cell **Order of Growth:** $O(V_i)$

Table 6.3: Analysis of computational cost and order-of-growth.

Technique	Computational Cost	Order of Growth
Image Rendering (2nd pass)	$C_r(G_i)$	rendering of texture-mapped image grid (+ texture filtering) **Order of Growth:** $O(V_i) + O(I)$
Non-Planar Projection Surfaces and Multiple Projectors for Non-Absolute Image Forming Systems (without transformations)	$V_i \cdot C_p(S_d) +$ $P \cdot C_r(G_i)$	projection of image-grid vertices onto display surface model + render projected image grid from perspective of each projector (+ texture filtering) **Order of Growth:** $O(V_i \cdot (S_d + P)) + O(P \cdot I)$
Non-Planar Projection Surfaces and Multiple Projectors for Absolute Image Forming Systems (without transformations)	$C_r(S) + C_{f2t}(I) +$ $C_{pt}(S_d) + P \cdot C_r(S_d)$	image generation + transfer of framebuffer content to texture memory + projective texture-mapping of image onto display-surface model + render display-surface model from perspective of each projector (+ texture filtering) **Order of Growth:** $O(P \cdot S_d) + O(P \cdot I)$ (we do not consider the scene complexity)
Multiple Viewpoints, Optical Chains, Non-Planar Projection Surfaces and Multiple Projectors for Curved Optical Elements	$VP\,(C_r(S) + C_{f2t}(I) +$ $E \cdot V_i \cdot C_{fl} +$ $V_i \cdot C_p(S_d))+$ $P \cdot C_r(G_i)$	Multiple viewpoints are required for stereo projection or to support multiple observers. *For each viewpoint:* generate image + transfer framebuffer content to texture memory + transform image grid through optical chain (E are the optical elements that lay on the optical path) + project image grid onto display surface model, render projected image grid from perspective of each projector (+ texture filtering) **Order of Growth:** $O(VP \cdot (I + V_i \cdot (E + S_d + P))) +$ $O(P \cdot I)$ (we do not consider the scene complexity)

Table 6.3: Analysis of computational cost and order-of-growth.

Technique	Computational Cost	Order of Growth
Multiple Viewpoints, Optical Chains, Planar Projection Surface and Single Projector for Planar Optical Elements	$VP(E \cdot V_{ts} \cdot C_{fl} + C_r(V_{ts}))$ if curvilinear transformations are required or $VP \cdot C_r(S)$ if all transformations are affine	Multiple viewpoints are required for stereo projection or to support multiple observers. *For each viewpoint:* transform scene vertices through optical chain (E are the optical elements that lay on the optical path) + render transformed scene. Note that if all transformations are affine, then $C_{fl} = 0$ and the sene does not have to be tessellated. **Order of Growth:** $O(VP \cdot E \cdot V_{ts})$ or $O(VP)$

Table 6.3: Analysis of computational cost and order-of-growth.

6.3 Performance Analysis for Curved-Optics Techniques

Several performance measurements have been taken from a number of experiments with the geometry-based and the image-based curved-optics techniques. These measurements are presented, discussed and compared within the following subsections. Note that we focus on curved-optics techniques because most of the rendering methods that support planar optics either apply affine transformations and consequently do not require additional computational cost (additional to regular pipeline rendering), or have a similar performance behavior as the tested techniques.

The same test platform has been utilized for all experiments that are described within this section: a 500Mhz Pentium III with 128MB RAM, nVidia GeForce II graphics, and Windows NT operating system. Note that the presented measurements strongly depend on the underlying hardware configuration.

6.3.1 Analysis of Geometry-Based Rendering Approach

The geometry-based reflected model-view transform described in section 4.2.1.1 maps image space vertices into projected object space vertices. Since the mirror optics was assumed to be curved, the transformation is curvilinear rather than affine.

Beside vertices, per-vertex properties (such as normal vectors, lighting and texture information, alpha values, etc.) might also have to be modified to achieve correct visual effects.

To efficiently simulate view-dependent global lighting effects within rendered scenes, Ofek [Ofe98, Ofe99] computes the shading values for unreflected vertices within the world coordinate system, and then assigns these values to the reflected vertices. In section 4.2.1.1, we mentioned that Ofek's method is inefficient for two reasons: First, the number of vertex transformations and shading computations increases with the number of scene vertices. To support smooth curvilinear transformations, however, the scene has to be highly tessellated. This results in a larger number of scene vertices and in a lower rendering performance. Second, since vertex transformations and shading are done explicitly, Ofek bypasses the standard rendering pipeline. Consequently, these computations cannot be supported by standard acceleration hardware.

Figure 6.1 illustrates the performance behavior of the geometry-based reflected model-view transform that computes per-vertex shading information explicitly.

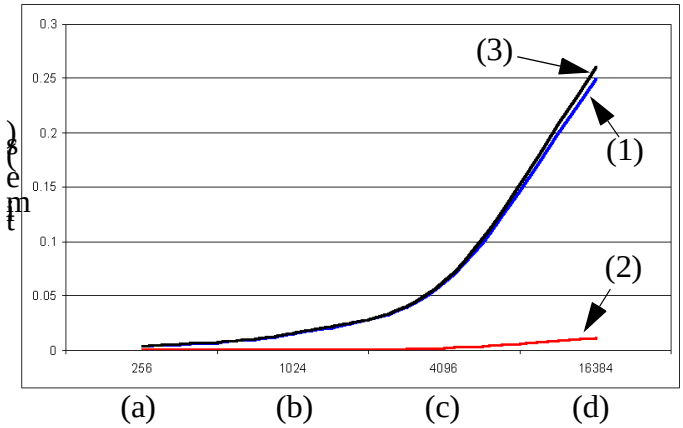


Figure 6.1: Geometry-based reflected model-view transform and explicit shading.

Rendering of different sample scenes (consisting of triangle meshes with: (a) 256 vertices connected by 450 triangles, (b) 1024 vertices connected by 1922 triangles, (c) 4096 vertices connected by 7938 triangles, and (d) 16384 vertices connected by 32258 triangles) has been timed to measure the computational cost of the vertex transformations and shading.
The blue curve (1) indicates the required time for the geometry-based reflected model-view transform and the per-vertex shading computations[26] while increasing the scene complexity. Both computations have been done explicitly (i.e., they have not been supported by acceleration hardware).
The red curve (2) indicates the required time for rasterization and per-fragment operations during rendering the transformed scene-triangles under consideration of the pre-computed per-vertex shading values. These computations did benefit form acceleration hardware, since they were carried out by OpenGL's rendering pipeline, which (in our case) was hardware-supported.
The black curve (3) finally indicates the total time (i.e., 1+2) that was required for each frame.
The diagram outlined in figure 6.1 shows that the corpus (~90-95%) of the time was spent for the explicitly vertex computations (i.e., transformation and shading), while rendering required only a fraction (~5-10%) of the total time.
Note that we assumed that the geometry-based reflected model-view transform is representative for all other per-vertex transformations that are introduced in chapter 4. This is sufficient because it shows approximately the same performance behavior than the other per-vertex transformations. In case that the refracted model transform (see section 4.1.3.) and/or the projected image transform (see section 4.1.4) are applied in addition, the time required for the vertex transformations increases respectively.

6.3.2 Analysis of Image-Based Rendering Approach

In this subsection, we want to present time measurements of the image-based rendering approach, discussed in section 4.2.2. By analyzing these measurements, we can find clues for

26.The ambient, diffuse and specular illumination components have been computed for each vertex to determine its total shading value. For this, a single point-light source has been assumed.

possible acceleration approaches. In addition, we compare them with the time measurements of the geometry-based rendering approach, presented in section 6.3.1.
Our image-based rendering approach consists of two rendering passes and intermediate image transformation steps. In the following, we want to analyze the performance behavior of the first and second rendering pass, as well as the image-based reflected model-view transform as a representative image transformation step. Note that in case additional transformation steps (e.g., the refracted image transform, described in section 4.2.2.4 and/or the implicit projected image transform, described in section 4.2.2.5) are applied, the time required for the image transformation increases respectively.
Figure 6.2 illustrates the performance behavior of the image-based reflected model-view transform and the two rendering passes. For these measurements, a constant image resolution (tw, th = 512x512 pixels) has been applied, while successively increasing the grid resolution (gw, gh). The scenery that has been rendered during the first pass was constant and consisted of a triangle mesh with 16384 vertices connected by 32258 triangles.

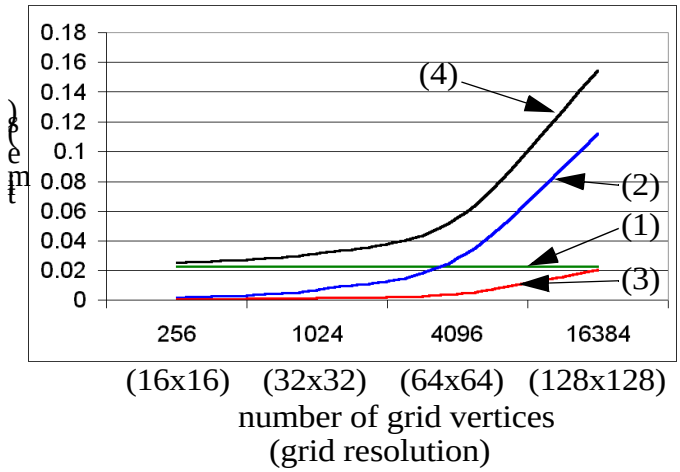


Figure 6.2: Image-based reflected model-view transform and the two rendering passes (with constant image resolution of 512x512 pixels and increasing grid resolution).

The green curve (1) indicates the time required for the first rendering pass (i.e., rendering of the scene and transferring the frame-buffer content to the texture memory). Since both -the scene complexity and the image resolution is constant, the time that was required for the first pass is also constant. Note that the entire first rendering pass was supported by hardware acceleration. Due to the relatively small scene that has been rendered, approximately 50% of this time was spent to transfer the frame-buffer content (i.e., 512x512x3= 786,432 bytes, for an RGB representation of the generated image) to the texture memory. Increasing either the scene complexity or the image resolution would increase the time required for the first rendering pass. The ratio of this time-rise depends strongly on the features of the applied acceleration hardware (i.e., polygon performance, memory bandwidth and transfer rate, etc.).
The blue curve (2) indicates the time that was required for the image-based reflected model-view transform. It increases together with the grid size. Note that the image-based reflected model-view transform was also not hardware supported. But in contrast to the geometry-based reflected model-view transform, the image transformation does not depend on the scene complexity. Consequently the time required for the image transformation is much smaller (due to

the much smaller number of vertices) than the time required for the scene transformation, and is constant (for a constant grid size).
The red curve (3) indicates the time required for the second rendering pass (rendering the transformed grid that is texture-mapped with the outcome of the first pass). As the first pass, the second pass is fully hardware supported and its performance also strongly depends on the features of the applied acceleration hardware (i.e., pixel fill rates for texture-mapping, rates for texture-filtering and compression, etc.). The required time for this pass increases slightly when increasing the grid size (due to the increased number of polygons to be rendered).
The black curve (4) finally shows the total time (i.e., 1+2+3) that was required for each frame. Compared to the corresponding measurements of the geometry-based approach presented in section 6.3.1, the image-based approach requires only a fraction (~10% for a grid size of 16x16 vertices - 60% for a grid size of 128x128 vertices) of the total rendering time. Note that in both cases, the same scene (a triangle mesh with 16384 vertices connected by 32258 triangles) has been rendered with the same resolution (512x512 pixels). However, the speed-up that is gained by using the image-based method increases drastically when increasing the scene's complexity.
Figure 6.3 illustrates measurements that were taken when applying a constant grid (gw, gh = 128x128 vertices) and successively increasing the image resolution (tw, th). The same scene has been rendered for this experiment as for the experiment described above.

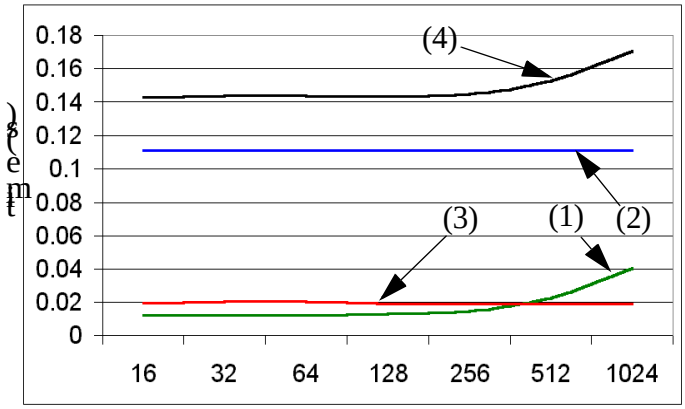


Figure 6.3: Image-based reflected model-view transform and the two rendering passes (with constant grid resolution of 128x128 vertices and increasing image resolution).

The green curve (1) again indicates the time required for the first rendering pass. Note that the time increases together with the image resolution, because this pass includes rasterization and the transfer of the frame-buffer content to the texture-memory.
The required time for the image-based reflected model-view transform is indicated by the blue curve (2). It is constant, since the grid resolution is constant.
The time required for the second rendering pass (indicated by the red curve (3)) is also approximately constant. This is due to the constant grid size and the constant number of linear interpolations that are required for the texture-mapping.
The black curve (4) shows the total time (i.e., 1+2+3) that was required for each frame and indicates that the best case approximately matches the worst case of the previous measurements (i.e., ~60% of the geometry-based approach's total rendering time). This can be attributed to an oversampled image grid.

The presented measurements imply that mainly two parameters can be modified to influence the rendering performance: the grid resolution and the image resolution.

6.4 Efficiency Analysis of Selective Refinement and Progressive Rendering

This section discusses the effectiveness of the selective refinement method that has been introduced in sections 4.3.1. We also evaluate the stability and responsiveness of the progressive refinement functions that have been described in section 4.3.2.2. Thus, the implications made in section 6.3.2 are taken into account.

6.4.1 Visual Appearance and Performance Analysis of Recursive Grid Refinement

We have introduced a recursive grid refinement algorithm that selectively refines the image geometry depending on the screen space error that is generated within the image space in subsection 4.3.1.

Our experiments were concerned with the performance of the different approaches described as well as with the resulting image quality in terms of the displacement error. Our Virtual Showcase mirror display that was driven by a Pentium IV with 2Ghz and FireGL II graphics acceleration was used to carry out these experiments.

Figures 6.4 and 6.5 illustrates several outcomes of the *RecursiveGridRefinement* algorithm for different image space error thresholds while using the same point of view, scene, and optics (i.e. the Virtual Showcase, as illustrated in figures 5.32-5.35). Note that stretched image regions with a lower curvature are rendered with a lower discrete LOD, and compressed regions with a higher curvature are rendered with a higher discrete LOD. Note also that LOD transitions are continuous. Figure 6.5 reveals how effectively the relevant image geometry can be identified with an oriented bounding box, rather than deforming the entire image grid.

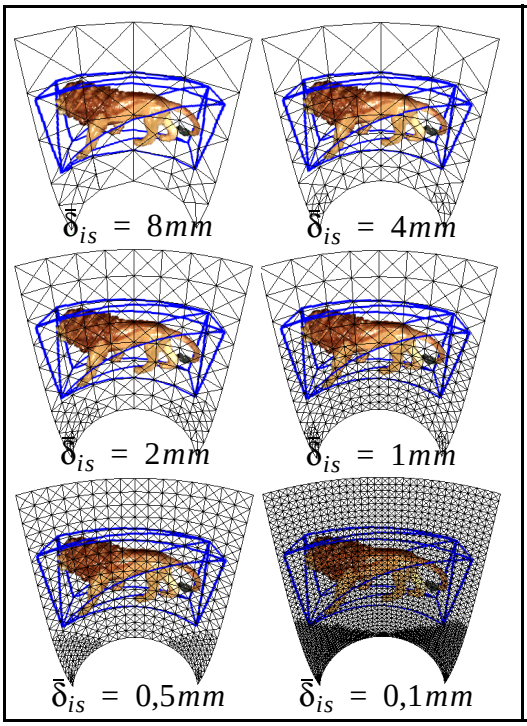


Figure 6.4: Selective grid refinement for different image space error thresholds.

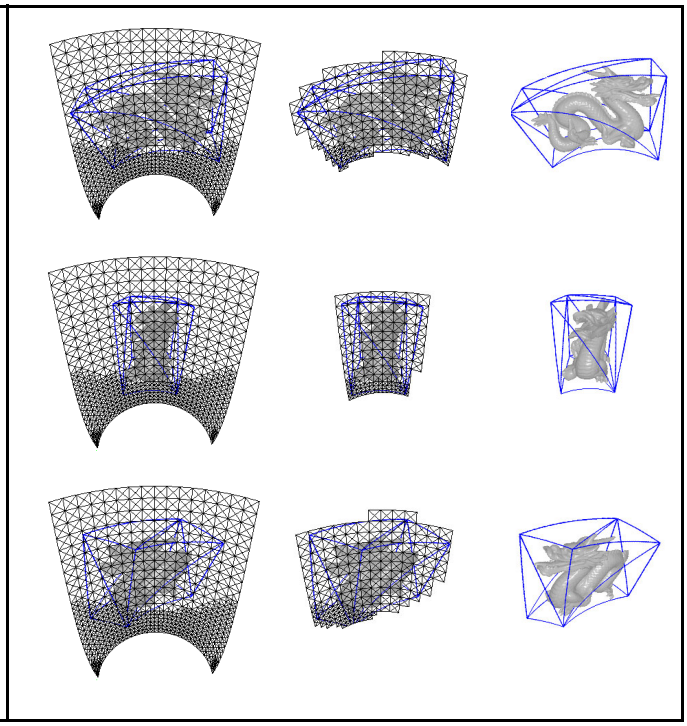


Figure 6.5: Spatially limited grids for different perspectives: entire refined grid (left), grid portion limited by container (center), resulting image (right).

We have chosen a Λ of 3 (i.e. error directions are only computed for the three lowest LODs), since our experiments have shown that this is a suitable value for our simple mirror geometry. The tessellation of the image grid varies with a moving viewpoint and is not necessarily as symmetric as shown in figures 6.4 and 6.5.
Figure 6.6 and figure 6.7 plot the reduction of vertex transformations and number of triangles to be rendered for the examples shown in figure 6.4.

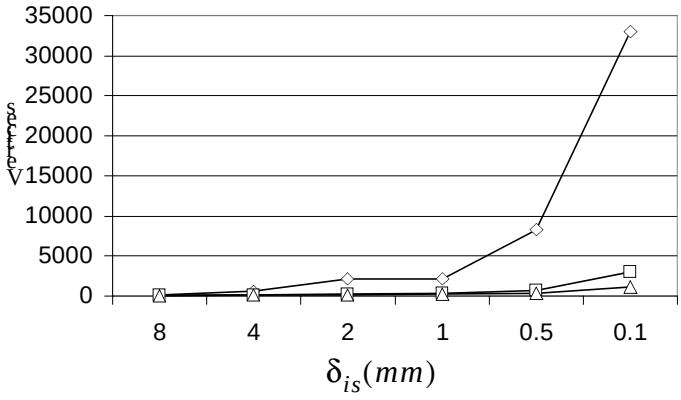


Figure 6.6: Number of vertex transformations for different $\bar{\delta}_{is}$.

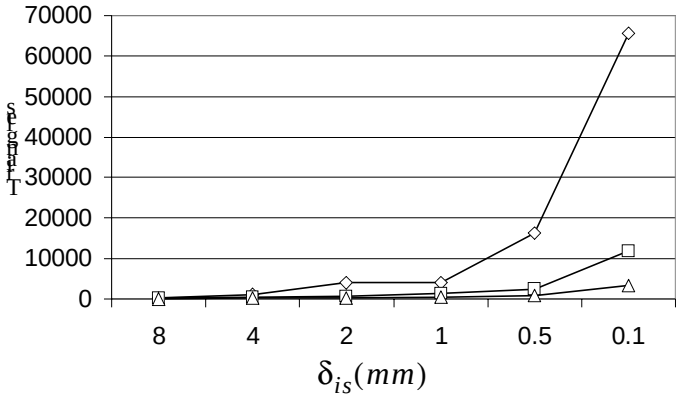


Figure 6.7: Number of rendered triangles for different $\bar{\delta}_{is}$.

Diamonds indicate measurements for the uniform image grid with a resolution that is appropriate to compensate for the required image space error. While squares represent measurements for the entire selectively refined image grid, triangles outline measurements for the spatially limited image portion.
Figure 6.8 illustrates the computation durations that were required for the image grid transformation and the rendering of the transformed grid.

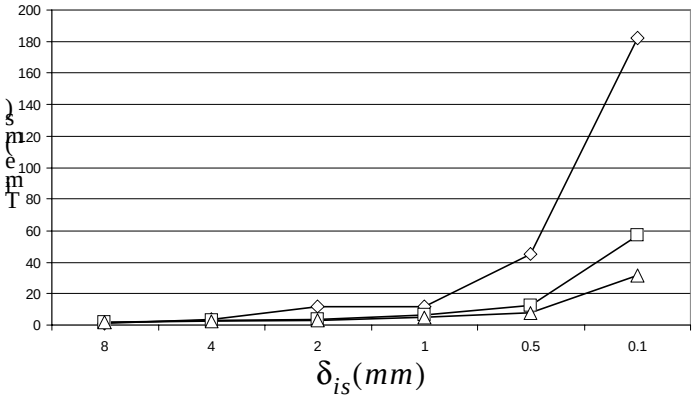


Figure 6.8: Transformation time + rendering time for different $\bar{\delta}_{is}$.

Table 6.4 presents the speedup factors for different image space errors while comparing the uniform grid method with the unconstrained selective refinement method (**A**) and with the spatially limited selective refinement method (**B**).

$\bar{\delta}_{is}$	*8mm*	*4mm*	*2mm*	*1mm*	*0.5mm*	*0.1mm*
A	0.88	1.31	3.05	1.79	3.70	3.19
B	0.86	1.51	3.94	2.44	5.78	5.78

Table 6.4: Speedup factor for different $\bar{\delta}_{is}$.

One can observe that for large error thresholds (e.g. 8*mm* in our examples) no speedup can be gained over the uniform grid method. For higher precision requirements, however, significant speedups are achieved. Note that the measurements include all required computations, such as recursions and minimizations.
Figure 6.9 shows an example as it can be observed in the Virtual Showcase mirror display which is illustrated in figures 5.32-5.35. Note that the photographs are not touched up. They are taken as seen from the observer's perspective. However, they have been rendered monoscopically. Note that the uniform (figure 6.9b) and the refined rectangular grid (figure 6.9c) are only shown to illustrate the effect of the selected image space error threshold. The image grid that approximates the bounding box's silluette (figure 6.9d) is actually produced by our algorithm. It can be noticed in figure 6.9c that the algorithm generates higher LODs for those image regions that are reflected by portions of the mirror which are less orthogonal to the observer. These regions generate a higher non-linearity within the warped image and would consequently produce a larger image space error. To prevent this, a higher geometric grid resolution is generated for these regions.

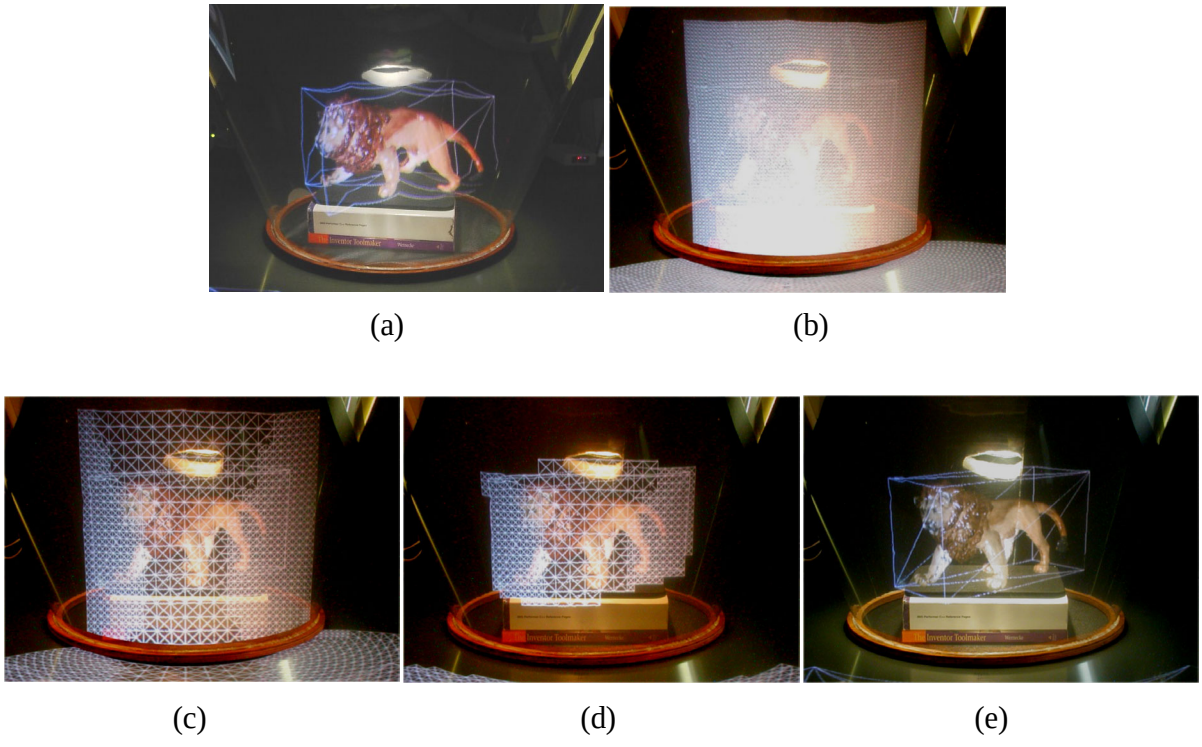


Figure 6.9: Results observed in the Virtual Showcase: (a) distorted image created with an undersampled uniform grid, (b) oversampled uniform grid, (c) selectively refined grid for $\bar{\delta}_{is} = 0{,}25mm$*, (d) grid portion that projects onto container, and (e) final result.*

Note that the grid geometry shown in figure 6.9b-e appears to be slightly noisy. This results from small bumps and dents that are randomly scattered over the mirror surface. They were caused by the irregular surface of the Plexiglas foil which was coated with a half-silvered mirror to built the optics. This distortion is too complex to be taken into account. Note also that the image in figures 6.9b,c is clipped by the boundaries of the mirror at its top and bottom. In the photographs, this clipping lets the image grid appear curved at these sides. However, it is rectangular.

6.4.2 Response and Stability Analysis of Progressive Refinement Functions

In section 4.3.2.2, we presented several refinement functions that modify specific parameters of the refinement criteria.

First, we discussed the refinement function used by Rusinkiewicz's QSplat rendering method [Rus00] (a concatenated square-root). We mentioned that this function approaches a predefined frame rate quickly and reacts to changes of the frame rate fast, but that it is difficult to keep a stable image quality for a continuously changing viewpoint or scenery, once the desired frame rate has been reached. We also said, that linear functions are not as responsive to frame rate changes, but provide a higher stability once the frame rate has been reached.

Figure 6.10 presents measurements that were taken when applying the two refinement functions separately. For these experiments, we tracked the modifications of the image space error threshold $\bar{\delta}_{is}$ (1) the current image resolution th, th (2), and the rendering time per frame T (3) of the first 60 frames. Note that all values are scaled to fit them into a single diagram.

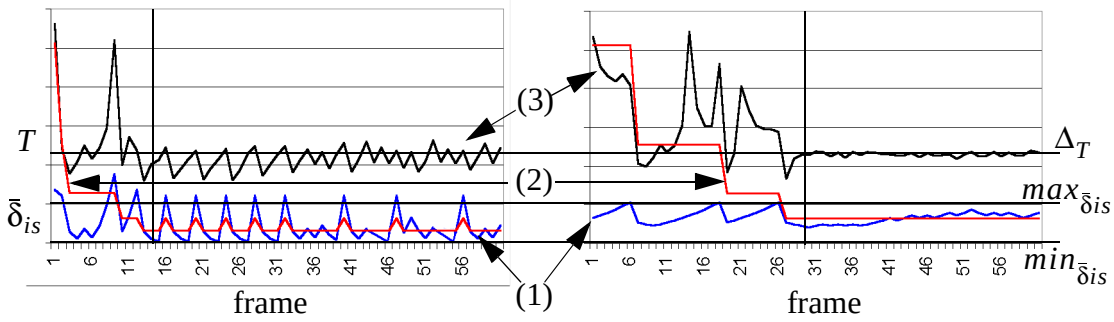


Figure 6.10: Refinement functions: $R_g = 1/\left(\sqrt{\sqrt{\bar{\delta}_{is}}}\right)$, $R_i = 2$ *(left) and* $R_g = 1{,}1$, $R_i = 2$ *(right).*

Figure 6.10-left illustrates the measurements for a concatenated square-root. We can see, that the desired frame rate Δ_T can be reached after approximately 15 frames. However, both -the image space error threshold $\bar{\delta}_{is}$ and the frame rate are unstable. Modifying the image space error threshold causes changes of the grid resolution. Each time $\bar{\delta}_{is}$ exceeds one of the two thresholds $max_{\bar{\delta}is}$ or $min_{\bar{\delta}is}$, the image resolution changes as well. Since both is continuously the case, the image quality changes continuously. These changes can be visually noticed by the observer as a ”pumping” of the image content.
Figure 6.10-right illustrates the measurements of a linear function. It takes approximately 30 frames to reach the desired frame rate, but the image quality remains stable once the frame rate has been reached. Note that the response and stability behavior of the linear refinement function can be modified by changing the function’s parameter (i.e., its constant value).
To support both -a fast response to frame rate changes and an appropriate stability of the image quality, we proposed the concatenation of linear and higher order refinement functions.
Figure 6.11 illustrates the measurements for the concatenation of the two functions, presented in figure 6.10.

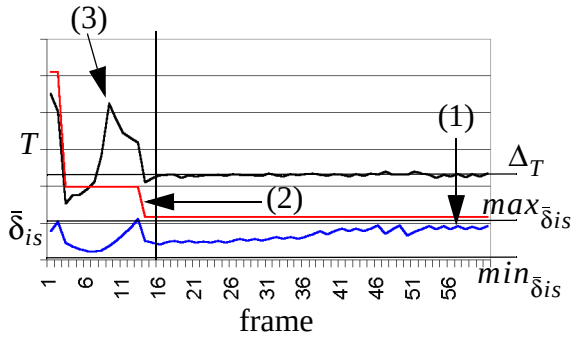


Figure 6.11: Concatenated refinement function: $R_{gf} = 1/\left(\sqrt{\sqrt{\bar{\delta}_{is}}}\right)$, $R_{is} = R_{gs} = 1{,}1$, *and* $R_{if} = \sqrt{\sqrt{[tw, th]}}$.

The desired frame rate is reached after approximately 15 frames and the image quality remains stable after the frame rate has been reached. How long it takes to render these frames depends strongly on the applied hardware, on the complexity of the scenery, and on the configuration of the image-based rendering method (e.g., the initial size of the base mesh and the initial resolution of the image). For the hardware that has been described in 6.3, an initial base mesh of 64x64 vertices, an initial image resolution of 512x512 pixels, and for the same scene that has been used for the previous experiments (a triangle mesh with 16384 vertices connected by 32258 triangles), it takes approximately 0.75s to render 15 frames. Note that due to the more liberal image resolution refinement function, a slightly higher final image resolution could be reached than in the other two experiments.

6.5 Parallel Processing Case-Study

This section presents a case-study for the parallel processing scheme, discussed in section 4.3.3. For the following experiments two identical PCs were configured as described in section 6.3, and directly[27] connected by a 100MBits/s Ethernet LAN.
To measure the performance of a distributed rendering, one computer has been set up as host and the other one as client. As described in section 4.3.3, the image generation step and the image transformation step were processed in parallel and were finally synchronized by transmitting the generated image from host to client and rendering it on the client site. Figure 6.12 illustrates the network performance that could be measured during the transmission of images of different sizes.

Figure 6.12: Transmission performance of a 100MBits/s LAN.

Note that the full bandwidth of 100MBits/s could not be reached to send the raw image data, because the TCP/IP packages that were used as carrier for the image data require some overhead (approximately 28% of the entire data stream, in our case).
Figure 6.13 illustrates two examples where the image transmission over the 100MBits/s LAN slows down the overall rendering performance so that parallel processing becomes impractical.

27.Not over intermediate routers.

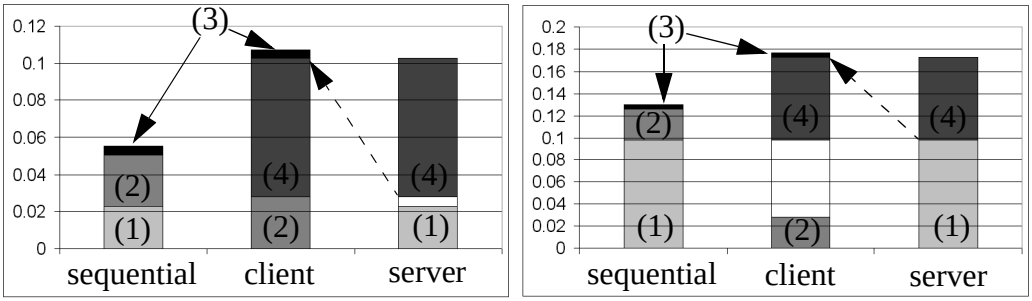


Figure 6.13: Sequential processing beats parallel processing.

Figure 6.13-left shows the sequential processing approach on the left hand side: First, a 512x512 image has been generated (1). The rendered scene contained 32k triangles. Then, a 64x64 image grid has been transformed (2) and the transformed image has been finally rendered (3). Note that for all experiments that are described in this section, the selective refinement or progressive rendering methods have not been applied. Consequently, all grid vertices have been transformed and the image resolution remained constant. For the parallel approach (right hand side), the server became idle (white block), since the image generation was faster than the image transformation. In this case, the transmission time (4) for the 512x512 image has been larger than the entire rendering time of the sequential approach.
For the experiment that is illustrated in figure 6.13-right, the number of triangles has been increased to 220k -simulating a more complex scene. Now, the large idle time on the client side makes the parallel approach inefficient.
Figure 6.14 shows an example where the rendering benefits from a parallel approach.

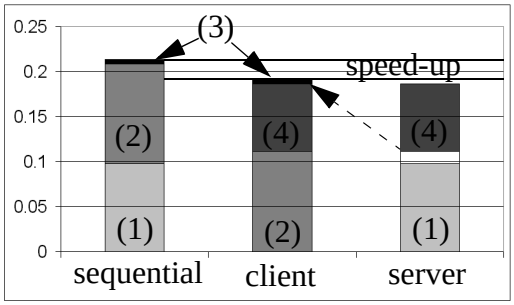


Figure 6.14: Parallel processing beats sequential processing.

In this case, the image grid has been increased to 128x128 vertices -balancing both processes and minimizing idle times.
In general we can say that a speed-up can only be gained by parallel processing if

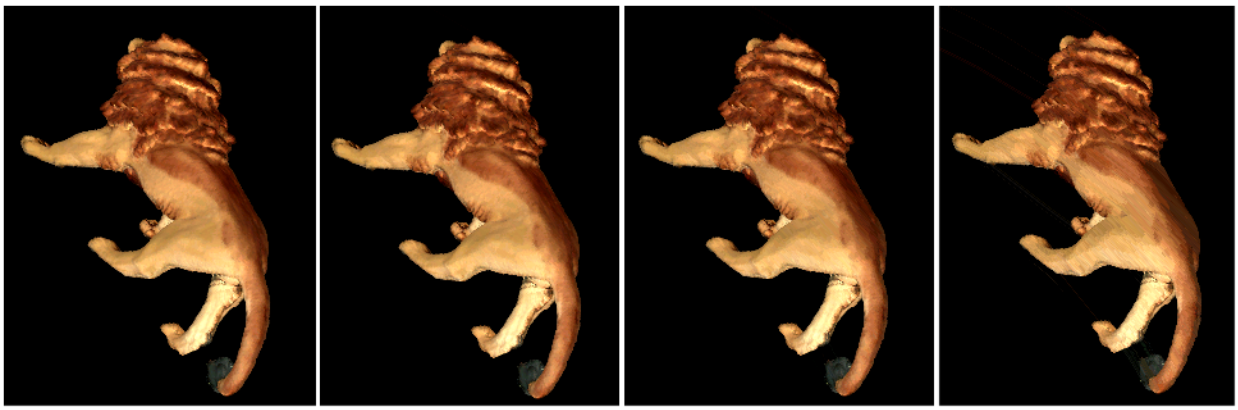
$$max(T_{ct}, T_{hg}) + T_t + T_{cr} < T_{sg} + T_{st} + T_{sr} \tag{6.1}$$

whereby T_{ct} is the time that the client requires to transform the image geometry, T_{hg} is the time that the host requires to generate the image, T_t is the image transmission time, T_{cr} is the time that the client requires to render the image, and T_{sg}, T_{st}, T_{sr} are the durations for the image generation, transformation and rendering on a sequential basis.

If the progressive rendering method is applied in combination with parallel processing, then the transmission time automatically influences the image quality in terms of being able to keep the desired rendering time on the client site (i.e., T_{ct}, T_{cr} and T_{hg} are adapted). This, in turn, impacts the transmission time (since the image resolution is modified) until the overall rendering time stabilizes (as illustrated in figure 6.11).
In summary, we can say that parallel processing can be beneficial if:

- A high-performance rendering host (such as a graphics workstation) is used for image generation at the front-end, because it is faster (i.e., $T_{hg} \ll T_{sg}$), or provides functionality that is not offered by a low-cost client which is used to drive a PBAR setup at the back-end;
- The transmission time is low (i.e., $T_t \rightarrow 0$);
- The two processes can be well balanced (i.e., $T_{ct} \sim T_{hr}$).

As mentioned in section 4.3.3, the transmission time can be reduced by encoding the image more efficiently before it is transmitted. Therefore, we tested a byte-wise adaptive Huffman coding algorithm [Knu85] and made the following observations: Although each image could be compressed down to approximately 30% of its original size and no additional information had to be transmitted, each coding step itself (i.e., encoding and decoding) required approximately 16x more time than the entire rendering process (including the transmission of the original image).
The problem specific coding methods that have been introduced in section 4.3.3 delivered better results:
Although for the *color-sequence-based method*, the encoded image contains some auxiliary bytes (such as positions and terminators), the elimination of black pixels provided a compression of down to 10-30% (depending on the image content) of the original image size during our experiments. In addition, encoding and decoding cost only a fraction of the overall rendering and transmission time (approximately 5-15%) and the image quality is not reduced.
The *run-length-based method* provides a higher compression (down to 5% in some cases) with similar coding/encoding performance as the color-sequence-based method. However, it reduces the image quality. We found, that a color-deviation threshold higher than 20 produces perceivable image differences and long runs of pixels appear as stripes (cf. figure 6.15-d).



(a) (b) (c) (d)

Figure 6.15: Run-length-based method. (a): uncompressed 1024x1024 texture, (b): 7% of (a) with $\overline{cd} = 10$, (c): 5% of (a) with $\overline{cd} = 20$, (d): 3% of (a) with $\overline{cd} = 40$.

Figure 6.15-d shows visual artefacts that result from color homogenization with a too large color-deviation threshold $\overline{cd}$.
The lion model that is outlined in figure 6.15 was used to generate the images for all of these experiments.
Other ways of reducing the transmission time are to apply faster network technology (figure 6.16 illustrates the estimated transmission rates of common 2 GBits/s fibre glass connections), or to use internal inter-process communication techniques (such as shared memory blocks) if a distributed processing on different machines is not required (i.e., if a single multi-processor machine is applied).

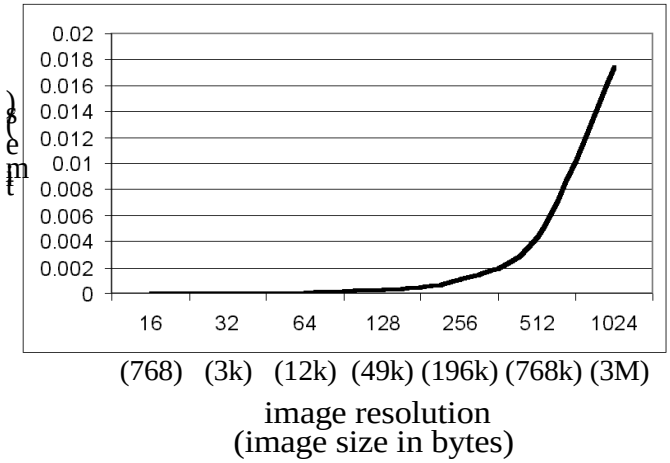


Figure 6.16: Estimated transmission performance of common 2GBits/s fibre glass connections.

6.6 Optical Characteristics and Visual Perception

The aim of this subsection is to give an impression of the optical characteristics of a selected PBAR configuration (primarily for its see-through mode). We therefore compare the Extended Virtual Table setup (see section 5.3) to the most commonly used type of AR display devices - Head-Mounted Displays. Note that the Extended Virtual Table has been selected, because its optical characteristics (especially its FOV-related properties) are comparable to those of HMDs. Most of the other proof-of-concept prototypes that have been introduced in chapter 5 are not supposed to offer a wide field of view, but they provide the same or similar resolution and ergonomics properties than the Extended Virtual Table.
For HMDs, the field of view is usually measured as the horizontal and vertical angles of the viewing volume that originates at the optical center and extends over the displayed image planes. Since for HMDs, the image plane is always centered with respect to the optical axis (i.e., it is on-axis), the FOV information is always constant (table 6.5 lists representative FOV information for low-budget and high-end HMDs). The image planes for our setups (i.e., the projection planes or their reflections) are always at a constant spatial position and alignment within the environment (the position and alignment of the reflected image planes also depend on the alignment of the mirrors). However, due to a changing viewpoint, they are not necessarily centered (i.e., they are off-axis). Thus, we cannot compute FOV information in the same way as it is done for HMDs. To be nevertheless able to compare the FOV-related characteristics of PBAR configurations with HMDs, we present a more general method to express the amount of our visual field in which computer graphics can be inserted:

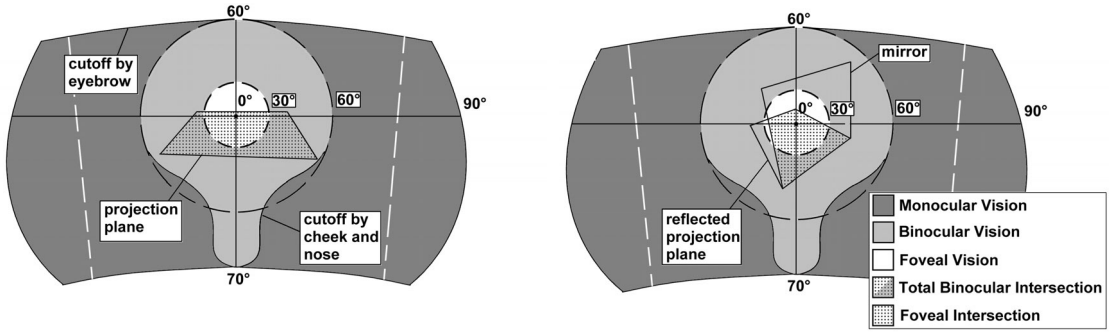


Figure 6.17: Total binocular coverage and foveal binocular coverage of the Virtual Table's projection plane (left) and of the reflected projection plane observed in the mirror (right).

To determine the area of the visual field in which graphics can be inserted and stereoscopically perceived, we compute the portion in which the image plane covers the binocular visual field (*total binocular coverage*). To be able to identify the area in which this graphics can be additionally perceived in focus, we also compute the portion of the foveal visual field that is covered by the image plane (*foveal binocular coverage*).
We approximate the binocular and foveal visual fields by determining the cones of sight that are defined by their FOV-angles and the viewer's viewing direction (determined from the head-tracker), not considering the asymmetrical shape that is formed by the cutoffs caused by the eyebrows, the nose or the worn goggles (e.g., HMDs or shutter-glasses). We first compute the areas of the two circular sections that result from slicing the cones of sight on any plane P_i that is perpendicular to the viewer's viewing direction. Then, the image plane that is observed from the current point of view is projected on P_i and the area of this planar projection is determined. To express the FOV-related characteristics of image planes that are not necessarily symmetrical or perpendicular to the viewing direction, we can compute the two ratios in which the projected image plane covers the binocular/foveal visual fields on P_i (i.e., we compare the area of the projected image plane with the two areas of the circular binocular/foveal sections). Note that even if the determined areas depend on the distance of P_i to the optical center, their ratios do not (i.e., the ratios are always constant on any sectional plane of the same viewing volume for the same image planes).
For instance, the binocular/foveal coverage of a 19" screen, viewed from distance of 24" (perpendicular to the viewer's viewing direction) is approximately 2.96%/26.63%. In general, the foveal fraction of the binocular visual field is 11.11% (total binocular coverage of the circular foveal section on P_i).

Display type	Resolution (pixels)	FOV at 100% overlap	Total binocular coverage (%)	Foveal binocular coverage (%)	Weight of goggles (grams)
Low-budget see-through HMDs	640x480, 800x600	36(H)x27(V)	3.31	29.80	600
High-end see-through HMDs	1024x768	64(H)x46(V)	11.26	93.61	2300
Virtual Table's Projection Plane	1280x1024,1600x1200/2	N/A.	Max.: 25.48 Min.: 18.07 Aver.: 22.02	Max.: 97.52 Min.: 43.72 Aver.: 70.66	… ≤ 93 (shutter glasses)
Reflected Projection Plane (Mirror at 80°)	1280x1024,1600x1200/2	N/A.	Max.: 7.66 Min.: 5.22 Aver.: 6.10	Max.: 66.71 Min.: 43.96 Aver.: 51.83	… ≤ 93 (shutter glasses)
Reflected Projection Plane (Mirror at 65°)	1280x1024,1600x1200/2	N/A.	Max.: 11.68 Min.: 7.61 Aver.: 9.26	Max.: 87.19 Min.: 44.55 Aver.: 66.60	… ≤ 93 (shutter glasses)
Reflected Projection Plane (Mirror at 50°)	1280x1024,1600x1200/2	N/A.	Max.: 15.20 Min.: 9.15 Aver.: 12.26	Max.: 93.00 Min.: 55.24 Aver.: 79.37	… ≤ 93 (shutter glasses)

Table 6.5: The Extended Virtual Table's optical characteristics compared to HMDs.

Table 6.5 lists the total binocular coverage and the foveal binocular coverage of low-budget and high-end HMDs, of the Virtual Table's projection plane[28] (cf. figure 6.17-left) and of the reflected projection plane under different alignments of the mirror (cf. figure 6.17-right). While the ratios of coverage are constant for the HMDs' fixed image planes, they dynamically change for setups with a non-constant (off-axis) alignment between viewpoint and image-planes. We therefore determined the highest and lowest ratios, as well as an average value for a moving viewer over time while the image-planes remain spatially constant. Note that this method is also applicable for other setups that cannot provide simple image planes or on-axis viewing (e.g., for hand-held devices or for systems with single/multiple or planar/curved projection planes).

28.The Virtual Table has been horizontally aligned.

The average binocular coverage of the reflected projection plane is usually smaller than the average coverage of the Virtual Table's projection plane. This is because the reflection might be cut off by the mirror (cf. figure 6.17-right) and it is always located further away from the observer than its real counterpart -which consequently causes a smaller projection on P_i. However, the binocular coverage of the reflection increases with more acute angles between mirror and projection plane and reaches a maximum at 45° (in this case, the projection plane and its reflection are perpendicular).
Obviously, the optical characteristics of the Extended Virtual Table converge to the parameters of high-end see-through HMDs. However, they depend on the applied projection technology (e.g., the resolution of the projector and the size of the projection plane) and on the optical extension (e.g., the size of the mirror). Since the projection technology is improving very fast (for example, in the area of high resolution multi-projector displays [Fun00]), we can expect much better results in the near future.
Another problem of HMDs that make use of a fixed focal length optics is that the focal distance of the image plane (which appears relatively close to the eyes) is not the same as the distance to real objects. Thus, the viewer has to alternatively shift focus between the image and real objects in terms of seeing both clearly at the same time. This problem is mostly ignored, since it influences image clarity and not the geometric placement of the superimposition. For PBAR configurations, however, the focal length to the reflected image plane (which appears relatively far away from the viewer) is not fixed and also changes with a moving viewer. Since the reflected image plane appears at the same spatial environment as the real objects, the problem of focus is much improved over HMD approaches - although it is not completely eliminated.
As for all semi-immersive projection-based VR systems, lag during image composition is not perceived as extreme with our system as with HMDs (especially for fast head movements). Like for semi-immersive projection-based VR systems, this prevents simulation sickness.
We can conclude that the optical characteristics of the Extended Virtual Table setup as a representative PBAR configuration can catch up with high-end optical see-through HMDs. Patrick, et al [Pat00] and Johnson and Steward [Joh99] indicate that statistically no significant difference in acquiring spatial knowledge can be found between closed-view HMDs and large projection screen conditions. However, they also state that the lower-cost projection screens are an attractive alternative to expensive and uncomfortable HMDs (i.e., discomfort that is due to the poor ergonomics and simulation sickness that is caused during fast head movements). These findings can also be applied to relate optical see-through HMDs to projection-based AR devices.

6.7 Summary of Evaluation Results

In this chapter, we presented several non-coherent evaluations of selected techniques and proof-of-concept prototypes. The results of this chapter are summarized below:
We compared the precision of our analytical refracted model transform approximation with a precise numerical method and concluded that the average deviation between precise numerical refraction model transform and analytical approximation is far below the average positional accuracy of the applied electro-magnetic tracking device. Thus, a higher distortion is caused by the inaccurate head-tracker than by applying the approximation to correct refraction misalignments. However, if refraction is not handled at all, the resulting optical distortion is higher than the one caused by tracking-errors. The analytical approximation required only a small fraction of the transformation time that is required by the numerical minimization.
The computational cost and the order-of-growth of all presented rendering and transformation techniques have been discussed in theory. In addition, we presented practical performance

measurements of the curve-optics techniques and provided experimental results of the selective refinement and progressive rendering acceleration schemes.
The comparison between the geometry-based approach and the image-based approach revealed, that for curved optics, the image-based method performs better than the geometry-based method. This can be attributed to that facts, that the corpus (~90-95%) of the rendering time of the geometry-based approach was spent for the explicit vertex computations (i.e., transformation and shading), and that the scenes had to be highly tessellated to support a smooth curvilinear transformation.
The presented measurements of the image-based approach implied that mainly two parameters can be modified to influence the rendering performance: the grid resolution and the image resolution.
The selective refinement algorithm that was used in combination with our multi-pass rendering method generates image grids with appropriate regional grid resolutions on the fly. On the one hand, the refinement algorithm prevents oversampling and texture artifacts. On the other hand, it speeds up rendering and image warping for such displays significantly while guaranteeing a maximal error on the image plane. Our experiments have shown that the speedup increases proportional to the precision requirements by applying the selective refinement algorithm instead of transforming the entire grid. This means that for the particular examples presented above, speedup factors of up to 6 could be achieved for a required precision of $\bar{\delta}_{is} = 0{,}5mm - 0{,}1mm$.
The performance measurements that have been presented above were taken under different hardware conditions. They were used to derive conclusions by comparing single values that have been taken on the same hardware platform for an individual experiment. However, the applied hardware improved over time (i.e., while carrying out the experiment of the different sections within this chapter) and did not always reflect the capabilities of today's graphics hardware. This is especially true for the experiments that are presented in sections 6.3 and 6.4.2. The hardware on which these experiments have been carried out was completely outdated and the complexities of the rendered scenes were unrealistically small. The measurements that are presented in section 6.4.1, however, were taken on a more modern hardware platform.
With figure 6.18, we present realistic overall measurements taken on a more modern test platform, different up-to-date graphics adapters and for a more complex scene. The test platform that has been used to take these measurements was a Dell Precision 530 Pentium IV with Xeon 4 processor (1.7 GHz) and 1GB PC800 RDRAM. The rendered scene was the lion model that is outlined in figure 6.15. The model consists of 183,408 vertices, connected by 367,277 triangles and contains per-vertex colors. The images of the model were generated with the QSplat point-based rendering method [Rus00]. After the first rendering pass, the images were transferred from the framebuffer into the texture-memory, they were deformed with respect to the viewpoint and the optics (selective grid refinement was enabled with a 64x64 base mesh), and they were finally rendered during the second pass.

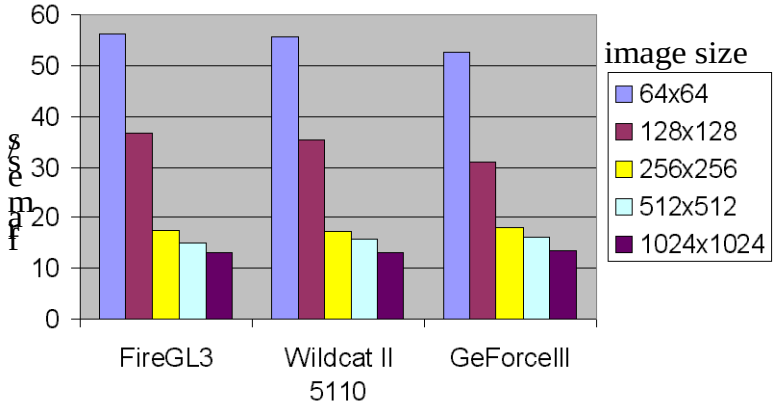


Figure 6.18: Performance measurements of the image-based method for state-of-the-art test platform and a realistic scene.

It is obvious that the overall performance of our method depends on the image resolution. This is because multi-pass rendering requires to transfer the image that has been rendered into the framebuffer during the first pass into the texture memory before it is displayed during the second pass. Although today's acceleration hardware is able to perform this entirely on the graphics board, it is still the most time consuming step of our approach. This step cannot be accelerated by software algorithms without decreasing the image quality.

To tackle this problem, we have developed a reactive progressive rendering algorithm that modifies both parameters (i.e., grid resolution and image resolution) during runtime. We have demonstrated that this method could quickly approach a desired frame rate and that it could keep it on a stable level, after it has been reached. However, decreasing the image resolution to increase the overall frame rate results in a loss of image quality.

Further, a parallel processing case-study was discussed and it was concluded that parallel processing can be beneficial if a high-performance host is used for image generation (e.g., because it is faster or provides functionality that is not offered by a low-cost client), the image transmission time is low, and the two processes can be well balanced.

To reduce the image transmission time, we have introduced two problem-specific image coding approaches that are optimized for optical see-through presentations. During our experiments, these methods provided a compression of down to 10-30% (without reduction of the image quality) or down to 5% (with reduction of the image quality) of the original image size. For both methods, the encoding and decoding times were only a fraction of the overall rendering and transmission time (approximately 5-15%).

Finally, we have shown that the optical characteristics (field-of-view and resolution) of a selected proof-of-concept prototype can catch up with high-end optical see-through HMDs. We have introduced a general method that allows for comparing field-of-view related characteristics of heterogeneous Augmented Reality displays that apply possible off-axis image planes.

7 Conclusion and Future Perspectives

7.1 Conclusion

This work proposes to combine the technological and ergonomic advantages of the well established projection-based Virtual Reality with the application potentials of Augmented Reality. The introduced projection-based Augmented Reality (PBAR) concept aims (taking pattern from the evolution of VR) to detach the display technology from the user and to embed it into the real environment instead. However, it is not intended to substitute other display concepts, such as head-attached displays, but rather to present an application specific alternative.
We define a PBAR configuration to be a spatial projection screen that is enhanced with optical see-through technology and supports stereoscopic, view-dependent and off-axis viewing of a graphically superimposed real environment. If a PBAR configuration supports a seamless combination of VR and AR by approaching a conceptual and technical extension of traditional Virtual Reality by means of Augmented Reality, we speak of extended Virtual Reality (xVR).
The basic principles of geometric optics represent the mathematical, physical and physiological foundations for this work. For our purposes, we assume that an optical system consists of four major components: mirrors, lenses, detectors (the human eyes in our case) and stereoscopic projection displays. Planar mirror beam-splitters that produce true stigmatic reflections between all object-image pairs, and convexly curved mirrors that represent non-absolute optical systems have been utilized in combination with spatial projection screens. Since the applied mirrors are half-silvered and simultaneously transmit and reflect light, they optically represent a combination between full mirrors and planar or parallel spherical lenses -reflecting and refracting light. Due to the limited retinal resolution of the eyes, small aberrations of non-stigmatic mappings caused by thin curved mirrors are not detected. However, larger aberrations caused, for instance, by refraction of thick lenses make an exact superposition of transmitted and reflected images impossible. For this reason, we describe a general analytical approximation of off-axis in- and in/out-refraction transformations. We proved that the on-axis refraction transformation for centered systems -which is commonly referred to in the optics literature- and Heckbert's paraxial approximation of the refraction transformation used for beam-tracing and derived techniques are special cases of our method. With the aid of one of our proof-of-concept prototypes, we have shown that the average deviation error between a more precise but complex numerical computation and our light-weight analytical approximation is far below the average positional accuracy of the applied tracking technology.
In general, three main achievements have been made with this work:

- The introduction and formulation of the PBAR and the xVR concepts;
- The introduction and implementation of novel rendering techniques for PBAR configurations that provide interactive frame-rates on low-cost off-the-shelf hardware;
- The introduction and realization of novel PBAR configurations that solve problems which are related to traditional VR/AR devices and open new application possibilities.

These achievements will be discussed below:

The introduced interactive rendering techniques can be applied for planar and curved optics. Depending on the utilized optical elements, these techniques have to neutralize physical reflection and refraction transformations of the projected virtual scene so that the optically formed images appear orthoscopic, stereoscopically and perspectively correct and undistorted to an observer. All four types of optical elements are taken into account: reflectors, refractors, projectors, and perceptors. Since planar mirrors are absolute optical systems, we are able to apply

affine model and viewpoint transformations for such elements. These transformations can be integrated into traditional rendering pipelines and consequently can be fully supported by hardware acceleration without causing additional computational costs. In contrast to this, lenses and curved mirrors require curvilinear transformations that individually modify the per-vertex properties of each scene element. We found that an image-based approach is more efficient and more flexible than a geometry-based approach for such optics. Our final two-pass image-based method avoids a direct access to the scene geometry, and consequently prevents the time-consuming transformations of many scene vertices. In addition, it is not restricted to a geometry-based first rendering pass, but rather supports any image generating rendering method. The method is proposed in form of a rendering pipeline that comprises a sequence of rendering and non-affine image transformation components. It can be easily configured and extended to support new optics and projection technology, and it is flexible enough to be smoothly integrated into existing software frameworks. Additionally, it takes as much advantage of hardware-implemented rendering pipelines as currently possible. To increase the performance of components that can presently not be supported by hardware acceleration we proposed several software acceleration schemes, such as selective grid refinement, progressive rendering, parallel processing and image coding. Specifically to support optics which requires to correct non-linear image distortion by applying multi-pass rendering and image warping, we introduced a novel algorithm that generates appropriate regional levels of detail instead of applying a uniform image geometry during runtime. This method allows to consider the error that is caused from a piecewise linear texture interpolation and to minimize it by adapting the underlying image geometry. On the one hand, the refinement algorithm prevents oversampling and texture artifacts. On the other hand, it speeds up rendering for such displays significantly while guaranteeing a maximal error on the image plane. The developed rendering techniques provide interactive frame rates with off-the-shelf and low-cost hardware, such as PCs.
To prove the feasibility of our rendering techniques and to outline possible applications of projection-based Augmented Reality configurations, we have introduced and realized a variety of proof-of-concept prototypes. The general objective of the PBAR concepts is to combine the advantages of the well established projection-based Virtual Reality with the application potentials of Augmented Reality. Traditional head-attached AR displays, such as HMDs, currently entail a number of technological and ergonomic shortcomings that can be attributed to an imbalanced ratio between heavy optics (that results in cumbersome and uncomfortable devices) and ergonomic devices with low image quality:

- A limited resolution;
- A restricted field of view;
- Visual perception issues that are due to a fixed focal length caused by a constant and head-attached image plane;
- An increased incidence of discomfort provoked by simulation sickness;
- A difficult user and session independent calibration;
- The cumbersomeness of heavy devices.

These limitations prevent the usage of such displays in a number of application areas. Compared to head-attached AR devices, the application of spatial projection displays for Augmented Reality tasks provides improved ergonomics, a theoretically unlimited field of view, a high and scalable resolution, and an easier eye accommodation [Ras98c].
Evaluating the Extended Virtual Table as a representative PBAR configuration has exposed the following results:

- Its optical characteristics (resolution and field of view) converge to the parameters of today's high-end see-through HMDs. However, since the projection technology is improving quickly (for example, in the area of high resolution multi-projector displays [Fun00]), we can expect much better results in the near future;
- Because the reflected graphics appear near their real world locations, perception issues that are caused by the fixed focal length problem are much improved over HMD approaches - although they are not completely eliminated;
- Due to less degrees-of-freedom, an easy user and session independent calibration of the display is given;
- As for all projection-based VR systems, lag during image composition is not perceived as extreme as with head-attached displays (especially for fast head movements). Like for projection-based VR systems, this prevents simulation sickness;
- The applied light-weight glasses provide an improved ergonomic behavior.

These findings apply to all PBAR-configurations[29]. In addition, the PBAR concept presents possible solutions to several problems that can be attributed to today's projection displays in general:

- The window violation problem linked to semi-immersive projection screens;
- The occlusion problem linked to rear-projection screens;
- The support of multi-user viewing and seamless surround view.

However, the following general shortcomings are related to PBAR configurations:

- Since optical see-through elements are applied, they cannot present mutual occlusion of real and virtual environments. Consequently, bright real surfaces overlaid with dark virtual objects optically emphasize the real environment and let the virtual objects disappear;
- They do not support mobile applications because of the spatially aligned optics and projection technology;
- In most cases, the applied optics prevents a direct manipulative interaction with virtual and real objects that are located behind the optics;
- The number of observers that can be supported simultaneously is restricted by the applied optics.

We also demonstrated that by combining the advantages of projection-based Virtual Reality with the application potentials of Augmented Reality, the PBAR concept opens new application possibilities for both -AR and VR:
The Reflective Pad and the Transflective Pad represent handheld devices that, in combination with rear-projection systems, offer stereoscopic viewing. This cannot be realized with today's portable devices which, for instance, utilize Plasma LCD screens. They are currently applied in combination with interactive scientific visualization (such as for medical data and other volumetric information).
The Extended Virtual Table and the Transflective Board achieve a conceptual and technical extension of traditional Virtual Reality by means of Augmented Reality (xVR), and a seamless integration of such technology into habitual work environments. Hybrid modeling and assembly is one of the main applications of the Extended Virtual Table. Thus, virtual mock-ups (VMUs) can be modeled above the virtual workbench and then be assembled to corresponding

29.Field of view may represent an application-specific exception (e.g., Virtual Showcases do not require a wide field of view).

physical mock-ups (PMUs) that are located on the real workbench. Combining VMUs and PMUs to hybrid mock-ups (HMUs) allows for a more realistic early design review and allows early refinements within the conceptual product design phase. In addition, a seamless transition between modeling, assembly and review processes provided by the Extended Virtual Table and the Transflective Pad helps to minimize media disruptions between different systems and technologies.
Virtual Showcases are compatible with traditional museum or other exhibition displays. They provide new ways of merging and exploring real and virtual content. Their support for multiple observers and for a seamless surround view allow a collaborative and natural exploration of presented artefacts. These properties are unique for today's projection display technology and for more than two viewers. Our interactive showcases represent a step towards ambient intelligent landscapes, where the computer acts as an intelligent server in the background and users can focus on exploring the exhibited content rather than on operating computers. Virtual Showcases open several new application possibilities for both domains -Augmented Reality and Virtual Reality- and are currently being applied and evaluated within different domains, such as cultural heritage, scientific visualization (e.g., [Bim02b]), training and rehearsal, edutainment, and early design review.

7.2 Future Perspectives

While specific aspects of future work have been discussed for each of the proof-of-concept prototypes individually, this subsection presents an outlook for the introduced rendering methods and upcoming enabling technologies from a broader point of view.

PBAR Rendering Techniques
Since the interface between the proposed rendering techniques and the application is minimal, they could be integrated into an intermediate display driver that would be located between the rendering hardware layer (i.e., the driver of the graphics board) and the application layer. Similar abstract driver layers are currently proposed. So called "Z-Buffer stereo" (e.g., [Nvid01b]) is one example that offers stereoscopic rendering on the driver level -even if this is not explicitly implemented by the overlaying application.
An efficient distributed or remote rendering allows to realize web-enabled PBAR configurations that can be employed as display front-ends. This is especially of interest for public PBAR displays, such as Virtual Showcases. To make distributed and remote rendering efficient, appropriate steaming techniques have to be developed. This implies, for example, that our image coding methods need to be expanded and improved to perform well on wide area networks. Therefore, existing standards, such as MPEG, should be considered since they already cover established streaming methods and other techniques and tools that can be useful for PBAR configurations.
A less general adaption of existing advanced rendering techniques, such as point-based rendering (e.g., QSplat [Rus00]) or interactive raytracing (such as Light Fields [Lev96] or Lumigraphs [Grot96]) for PBAR configurations may reveal additional performance sources. This needs to be further investigated.

Diverging Applications for PBAR Rendering Techniques
The presented rendering techniques could also be beneficial for existing non-PBAR configurations. McKay's varifocal mirror system [Mck99a, Mck99b], for instance, applies additional lenses to stretch the reflected image and to support an autostereoscopic viewing for a static viewpoint. Yet, he does not employ any distortion correction to the displayed images. McKay reports, that his system suffers from spherical aberrations when the mirror exceeds a specific

curvature, and because of this only 80% of the mirror aperture can be used for imaging. Although McKay's system is centered, he would be able apply our rendering techniques with a fixed viewpoint to pre-warp the generated images with respect to the actual mirror curvature.

The introduced selective refinement algorithm can also be applied to neutralize optical distortion or to display undistorted images onto non-planar surfaces in general. Beside PBAR configurations other curved displays, such as cylindrical or spherical screens, can benefit from this approach. Today, multi-pass rendering in combination with projective textures is normally applied to correct distortion caused by such or similar displays (e.g, [Vanb00, Ras01, Ban01, Yan01]). These methods apply a constant underlying image geometry do not analyze or minimize the error that is caused by a piece-wise linear texture interpolation. To avoid visible texture artefacts highly tessellated image geometries are applied instead to increase the number of sample points and to shorten single interpolation paths. This, however, results in low overall framerates. In contrast to this, our refinement algorithm prevents oversampling and texture artifacts. It speeds up rendering and image warping for such displays significantly while guaranteeing a maximal error on the image plane. The adaption of the display specific components that are required by our algorithm to support non-PBAR displays represents an interesting future direction of extending the scope of our rendering techniques.
Our general analytical approximation of off-axis in- and in/out-refraction transformations could also be applied for rendering refractions as view-dependent global lighting phenomena within graphical 3D scenes. In appendix A, we gave evidence that Heckbert's paraxial approximation of the refraction transformation used for beam-tracing [Hec84] and later approaches that are based on Heckbert's method (such as [Die96, Die97]) are special cases of our method. We said that Ofek [Ofe99] applies an affine shear transformation to simulate refraction caused by curved surfaces. As criticized by Diefenbach [Die96], this type of transformation is a poor approximation of refraction -even if only paraxial rays are considered. To simulate off-axis refraction situations within graphical 3D scenes, our method generates more realistic results than the previous methods. The computational cost of our method, however, is comparable to the one of Ofek's virtual object method.

Enabling Technologies
While regular rendering passes and affine scene transformations can be completely executed by graphics accelerators of today's graphics adapters, intermediate per-vertex transformations are not yet supported by prevalent rendering pipelines (such as the one implemented in OpenGL). Consequently, they cannot benefit from current geometry acceleration hardware. At present, we have realized these transformations in software (i.e. they tax the main CPU and memory). Their hardware-realization is a next logical step. Thus, we can follow two general approaches: First, special-purpose acceleration hardware can be realized that implements a per-vertex transformation pipeline and complements off-the-shelf graphics hardware. Re-configurable computing technology, such as Field Programmable Gate Arrays (FPGA) [Vil97] seems to be appropriate for this purpose. Nevertheless, additive solutions that are manufactured in small numbers will be expensive, and will therefore hinder a wide market acceptance. Second, wide-spread and low-cost graphics hardware could be extended. Next generation graphics engines (e.g., [Lin01, Nvid01a]) support programmable per-vertex operations, which will allow hardware acceleration of the required per-vertex transformations. However, this new functionality is not yet standardized and is realized differently by each manufacturer. To avoid implementing many individual solutions that benefit from the support of different graphics boards, an abstract and standardized interface is required.
As described in section 6.7 the overall performance of our multi-pass method depends on the image resolution. This is because multi-pass rendering requires to transfer the image that has

been rendered into the framebuffer during the first pass into the texture memory before it is displayed during the second pass. Although today's acceleration hardware is able to perform this entirely on the graphics board, it is still the most time consuming step of our approach. This step cannot be accelerated by software algorithms without decreasing the image quality. However, tomorrow's graphics boards will allow to directly render into the texture memory. In fact, several hardware vendors recently announced that near-future releases of their drivers and graphics boards (e.g., [Nvid01a]) will support this. The time-consuming transfer of large image date from one buffer into another becomes superfluous. This will significantly speed up our multi-pass rendering techniques.

To efficiently support simultaneous rendering for multiple viewers on cost-effective rendering platforms (e.g., PCs), a networked cluster of rendering nodes could be applied. Each node would carry out the image generation and deformation for a single observer. However, in terms of displaying the final images on arbitrary locations of a single projection screen, they need to be composed within one image. One possibility for doing this is to utilize coding and streaming technology to transmit the images to a central node that is responsible for their composition and the presentation of the resulting image. However, the transmission of the images over an all-purpose network (e.g., an Ethernet) as well as their explicit composition might represent bottlenecks that slow down the overall rendering performance significantly. Recent advances in hardware image composition technology solve this problem. Instead of connecting the rendering nodes via an all-purpose network, they can be connected to a specific display subsystem. These subsystems allow the image data generated by the connected nodes to be mapped to any location of the connected displays without losing much time for transmission and composition. The latest devices (e.g., the Lightning-2 device [Stol01]) are able to scale in both dimensions -the number of rendering nodes and the number of displays. Such technology also supports to efficiently drive high-resolution displays that are built from a collection of multiple display units (e.g., projectors). As mentioned above, a high and scalable resolution is one of the technological advantages of PBAR configurations over head-attached displays.

To overcome the problem of mutual occlusion that can be related to all optical see-through displays (i.e., also to PBAR configurations), the integration of a light-blocking technology to selectively block the transmitted light and to highlight the reflected light might present a possible future improvement for PBAR setups. Thus, we can envision to integrate such technology directly into the optics (as proposed by Kiyokawa et al [Kiy00]) and to extend our rendering concept respectively. Instead of using light-blocking elements, however, we can also envision to selectively control the light on a per-pixel basis to solve the mutual occlusion problem. Thereby, we can replace the diffuse light bulbs that are used to illuminate the real environment with a single or multiple projector(s) (technically similar, but conceptually different as described in [Ras01]). The rendering pipeline that has been developed for PBAR configurations can be extended to incorporate a projector-driven static or view-dependent illumination that generates shadows at those areas on the real objects' surfaces where graphics is overlaid [Bim02a].

In general, we want to annotate that upcoming and new technology will not only open new possibilities for projection-based Augmented Reality, but also for other display concepts, such as head-attached displays.

Organic Light Emitting Diodes (OLEDs) [How01], for instance, may replace the crystalline LEDs that are currently being used to build the miniature displays for HMDs. OLEDs promise to produce cheap and very high-resolution full-color matrix displays that can give head-attached displays a technological push.

A variant of OLEDs are Light Emitting Polymers (LEPs) [Bur90] that provide the opportunity for the fabrication of large, flexible, full-color emissive displays with a high resolution, a wide

viewing angle and a high durability. Transparent LEPs may present future alternatives for PBAR configurations. However, LEPs have not yet left the basic research stages and will not be applicable to build stereoscopic AR displays in the near future.
In the short run, especially high-resolution and bright projection devices, high-performance and cost-efficient rendering hardware, reliable and precise tracking technology, and advanced interaction techniques and devices will pave the way for forthcoming PBAR configurations. However, the underlying technology must be robust, flexible and the technology that directly interfaces to users should adapt to humans, rather than forcing users to adapt to the technology. Therefore, human-centered and seamless technologies, devices and techniques will play a major role for projection-based Augmented Reality.

Appendix A

Refraction Approximation of Beam-Tracing

Heckbert and Hanrahan [Hec84] approximate refraction transformations by only considering paraxial rays (i.e., entering light rays that are exactly or nearly perpendicular to the refracting plane). This corresponds to the approximated on-axis refraction transformation for centered systems, which is a special case of our method (see section 2.4.2).
In this section, we want to prove that Heckbert's approximation is also a special case of our general method.
Heckbert assumes that for paraxial rays, objects seen through a polygon with the refraction index η appear to be η times their actual distance, since light travels slower in denser materials by precisely this factor. Therefore, he applies a scaling transformation that is perpendicular to the plane. This transformation (re-written into our notation) is given by (taken from [Hec84]):

$$\vec{p}_i = \vec{p}_o + (\eta - 1)(L\vec{p}_o)\vec{n} \tag{A1.1}$$

where $L = [a, b, c, d]$ and $\vec{n} = [a, b, c]$ are the parameters of the polygon's plane. If we substitute L in equation A1.1, we receive:

$$\vec{p}_i = \vec{p}_o + (\eta - 1)(\vec{n}\vec{p}_o + d)\vec{n} \tag{A1.2}$$

Note that this refraction transformation corresponds to the reflection transformation described in equation 2.10. The only difference is the factor of the second term, which is -2 for reflection and $(\eta - 1)$ for refraction. The reason for this is that reflection is simply a paraxial refraction with $\eta = -1$, as discussed in section 2.1.2.
Consequently, Heckbert can define a homogeneous 4x4 transformation matrix, similar to the one presented by equation 2.11 (taken from [Hec84]):

$$R_f = \begin{bmatrix} 1 + (\eta - 1)a^2 & (\eta - 1)ab & (\eta - 1)ac & (\eta - 1)ad \\ (\eta - 1)ab & 1 + (\eta - 1)b^2 & (\eta - 1)bc & (\eta - 1)bd \\ (\eta - 1)ac & (\eta - 1)bc & 1 + (\eta - 1)c^2 & (\eta - 1)cd \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{A1.3}$$

To prove that equation A1.2 is a special case of our general method, we have to show that our off-axis refraction transformation described in section 2.4.2 equals equation A1.2 for the special case that $\alpha_i = 0$.
As we can see in equation 2.16, we apply a translation transform perpendicular to the refracting plane to simulate refraction. This corresponds to equation A1.2, since $\vec{p}_o$ is translated along the normal $\vec{n}$ by the amount $(\eta - 1)(\vec{n}\vec{p}_o + d)$. Because $t = f(\vec{p}_o) = \vec{n}\vec{p}_o + d$, as described in section 2.4.2, we can substitute and receive:

$$\vec{p}_i = \vec{p}_o + (\eta - 1)t\vec{n} \tag{A1.4}$$

Note that in section 2.4.2, we are only interested in the absolute value of t and not in its location relative to the plane.
With respect to equations 2.16 and 2.18, we now have to prove that the factor $(\eta - 1)t$ is equivalent to our Δ for the special case that $\alpha_i = 0$.
If we solve equation 2.18 for $lim(\alpha_i \to 0)$, we receive:

$$\Delta = t(1 - \eta) \tag{A1.5}$$

Note that we use equation 2.2 with $\eta = \frac{\eta_1}{\eta_2}$, as it has been done by Heckbert.
Since Heckbert uses the normal vector that is inverse to ours, a positive t in Heckbert's notation would actually be negative with our normal representation. Consequently, we have to multiply equation A1.5 by -1 to convert it to Heckbert's notation:

$$\Delta = t(\eta - 1) \tag{A1.6}$$

We can conclude that Heckbert's paraxial approximation of the refraction transformation (which corresponds to approximations for on-axis centered optical systems) is a special case of our general off-axis refraction transformation. However, this approximation is limited to paraxial rays and on-axis situations. As noted by Heckbert, this is due to the limitations of beam-tracing to perform only linear transformations.

Appendix B

OpenGL's Transformation Pipeline

Figure B.1 illustrates OpenGL's transformation pipeline as it is presented by Neider et al [Nei93]. It is a component of OpenGL's rendering pipeline which is also discussed in [Nei93].

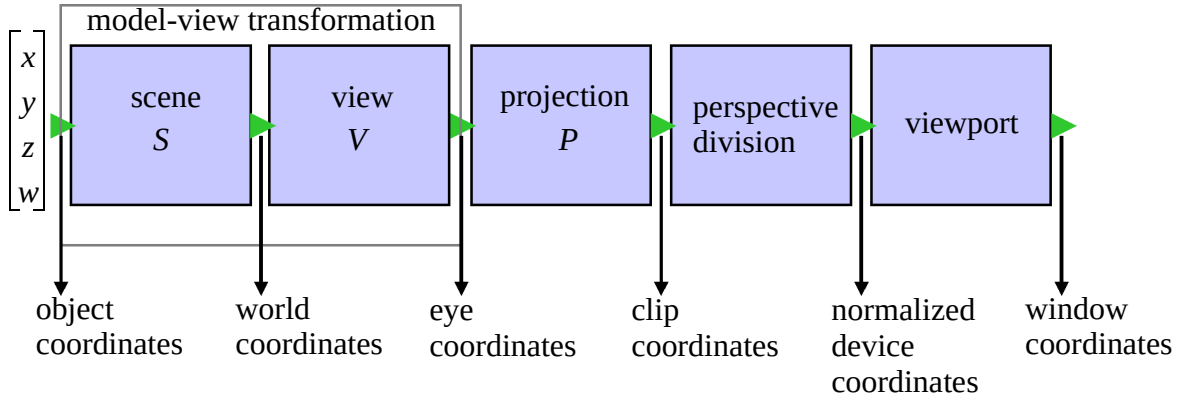


Figure B.1: OpenGL's transformation pipeline.

Object vertices (represented as homogeneous coordinates) are multiplied by the transformation matrix which is a composition of the *model-view matrix* and the *projection matrix*. These transformations are represented as 4x4 homogeneous matrices.
The *model-view transformation* can be conceptually splitted into the *scene transformation* and the *view transformation.*

The *scene transformation matrix S* can be composed of a sequence of other 4x4 transformation matrices which are all multiplied onto the *model-view matrix.* OpenGL offers a number of scene transformation commands which are all translated into corresponding 4x4 matrices and multiplied with the current model-view matrix. Examples for scene transformation commands are glTranslate(), glScale() and glRotate(). The corresponding matrices can be found in [Nei93]. The scene transformation maps the vertices from the object coordinate system into the world coordinate system.

The *view transformation matrix V* is either generated by an appropriate composition of glTranslate(), glScale() and glRotate() commands, or an additional utility library is used that provides an explicit view transformation command which "wraps" the basic commands. These commands are also translated into corresponding 4x4 matrices and multiplied onto the model-view matrix. An example is the gluLookAt() command, provided by the OpenGL Utility Library (GLU) -an extension of OpenGL. The view transformation maps the vertices from the *world coordinate system* into the *eye (or camera) coordinate system*.
Note that the inverse transpose mode-view transformation matrix is automatically applied to the surface normal vectors.

The *projection transformation matrix P* is generated by calling an appropriate projection command. OpenGL offers two commands: glFrustum() (or gluPerspective from the GLU) for a perspective projection and glOrtho() for an orthographic projection. They generate the corresponding transformation matrix and multiply it onto the projection matrix. The corresponding matrices can be found in [Nei93]. The projection transformation first maps the vertices from the eye coordinate system into the *clip coordinate system*, and then into the *normalized device*

coordinate system after applying the *perspective division* (i.e., $\left[\frac{x}{w}, \frac{y}{w}, \frac{z}{w}\right]$). While the clip coordinates are in the range $[-w,w]$, the device coordinates are normalized to the euclidean space $[-1,1]$. The normalized z component is used for depth handling, and is not effected by the viewport transformation.

The *viewport transformation* maps the $[x_{nd},y_{nd}]$ components of the normalized device coordinates into the *window coordinate system*. OpenGL provides the glViewport() command to support this mapping. A homogeneous matrix is not generated for this operation. Rather than that, the scaling transformation is applied explicitly to the $[x_{nd},y_{nd}]$ components of the normalized device coordinates. The window coordinates are normalized to the range $a \le x_w \le (a + width), b \le y_w \le (b + height)$, where $[a,b]$ is the lower left corner and $width$, $height$ are the dimensions of the window in pixels. The window coordinates are computed from the normalized device coordinates as follows:

$$x_w = (x_{nd} + 1)\frac{width}{2} + a \quad \text{and} \quad y_w = (y_{nd} + 1)\frac{height}{2} + b \tag{B.1}$$

Appendix C

Off-Axis Projections with OpenGL

Figure C.1 illustrates an off-axis projection situation. To determine the corresponding viewing frustum, the scene's bounding box (BB) is required to compute the near ($near$) and far (far) clipping planes.

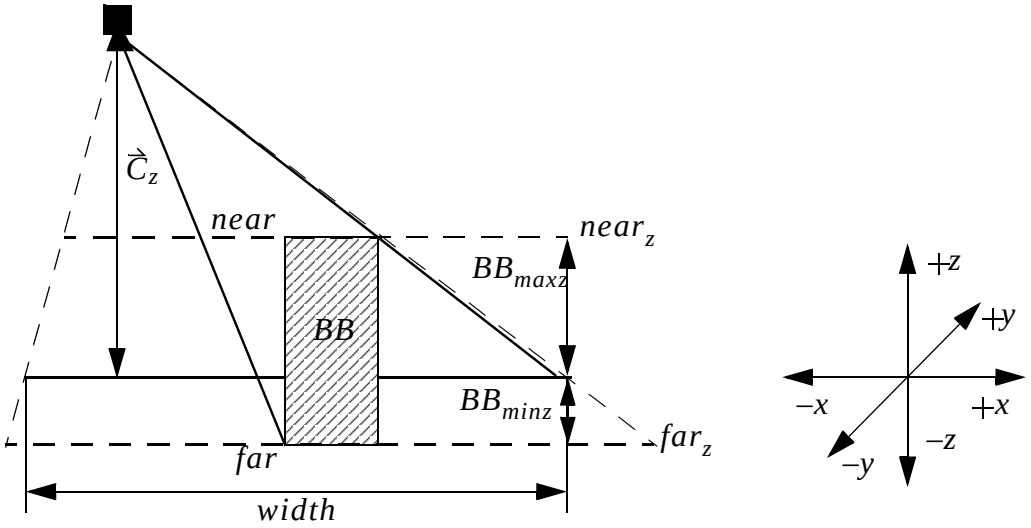


Figure C.1: Off-axis projection.

Give the position of the camera $\vec{C}$, the near and far clipping planes are computed as follows:

$$near_z = \vec{C}_z - BB_{maxz} - 1 \;\;,\;\; far_z = \vec{C}_z - BB_{minz} + 1 \tag{C.1}$$

Note that an offset of ± 1 is required to prevent clipping of the front or backside faces which are aligned with the bounding box's top or bottom.

Assuming an on-axis situation (i.e., the camera is located at $\vec{C} = [0, 0, \vec{C}_z]$), the size of the near clipping plane can be determined with a simple triangulation:

$$near_{width2} = \frac{near_{width}}{2} = \frac{width}{2} \cdot ratio \,,\; near_{height2} = \frac{near_{height}}{2} = \frac{height}{2} \cdot ratio \tag{C.2}$$

where $width, height$ are the dimensions of the projection plane or drawing window (in x, y directions), and ratio is given by:

$$ratio = \frac{near_z}{\vec{C}_z} \tag{C.3}$$

Since the projection is off-axis, a projectional shift has to be computed before the parameters of the viewing frustum can be determined:

$$shift_x = \vec{C}_x \cdot ratio \quad , \quad shift_y = \vec{C}_y \cdot ratio \tag{C.4}$$

Using OpenGL's glFrustum() function and GLUs gluLookAt() function [Nei93], the viewing frustum can be set as follows:

ProjectOffAxis($\vec{C}$, BB, $width$, $height$ ***):***

```
1:  glMatrixMode(GL_PROJECTION);
2:  glLoadIdentity();
3:  glFrustum(−near_width2 − shift_x , near_width2 − shift_x ,
4:                −near_height2 − shift_y , near_height2 − shift_y ,
5:                near_z , far_z );
6:  glMatrixMode(GL_MODELVIEW);
7:  glLoadIdentity();
8:  gluLookAt(C_x , C_y , C_z , C_x , C_y , 0, 0, 1, 0 );
```

Bibliography

[3M01] 3M, CORP. [P-18 is the most efficient Scotchtint Sun Control film. *Retrieved from the World Wide Web:* http://www.mmm.com/market/construction/html/products/product73_p.html, 2001.

[Agr97] Agrawala, M., Beers, A. C., Fröhlich, B., McDowall, I., Hanrahan, P., and Bolas, M. The Two-User Responsive Workbench: Support for Collaboration Through Individual Views of a Shared Space. *Computer Graphics (proceedings of SIGGRAPH'97),* pp. 327-332, 1997.

[Art01] Art+Com, AG, *Retrieved from the World Wide Web:* http://www.artcom.de/projects/vrf/welcome.en, 2001.

[Arth98] Arthur, K., Preston, T., Taylor, R., Brooks, F., Whitton, M., and Wright, W. Designing and Building the PIT: a Head-Tracked Stereo Workspace for Two Users. *In proceedings of 2nd International Immersive Projection Tecnology Work shop (IPT'98),* 1998.

[Arv01] ARVIKA - Augmented Reality in Industrial Applications, *Retrieved from the World Wide Web*: http://www.arvika.de, 2001.

[Arvo86] Arvo, J. Applications of irradiance tensors to the simulation of non-lambertian phenomena. *Computer Graphics (proceedings of SIGGRAPH'86),* pp. 335-342, 1986.

[Asc01] Ascension Technology, Corp., Flock of Birds. *Retrieved from the World Wide Web:* http://www.ascension-tech.com/products/flockofbirds/flockofbirds.htm, 2001.

[Azu95] Azuma, R. Predictive Tracking for Augmented Reality. *Ph.D. dissertation, University of North Carolina at Chapel Hill*, TR95-007, 1995.

[Azu97] Azuma, R. T. A Survey of Augmented Reality. *Presence: Teleoperators and Virtual Environments*, vol. 6, no. 4, pp. 355-385, 1997.

[Ban01] Bandyopadhyay, D., Raskar, R., and Fuchs, H. Dynamic Shader Lamps: Painting on Real Objects. *In proceedings of International Symposium on Augmented Reality (ISAR'01), pp. 207-215, 2001.*

[Baj92] Bajura, M., Fuchs, H., and Ohbuchi, R. Merging Virtual Objects with the Real World: Seeing Ultrasound Imagery Within the Patient. *Computer Graphics (proceedings of the ACM SIGGRAPH '92),* Annual Conference Series, pp.203-210.

[Bar01a] Barco, Inc. Baron. *Retrieved from the World Wide Web:* http://www.barco.com/projection_systems/products/product.asp?GenNr=324, 2001.

[Bar01b] Barco, Inc. Consul. *Retrieved from the World Wide Web:* http://www.barco.com/projection_systems/products/product.asp?GenNr=325, 2001.

[Bas99a] Bastos, R., Hoff, K., Wynn, W., and Anselmo, L. Increased photorealism for interactive architectual walkthroughs. *In proceedings of Symposium on Interactive 3D Graphics*, pp. 183-190, 1999.

[Bas99b] Bastos, R. Superposition Rendering: Increased Realism for Interactive Walkthrough. *Ph.D. dissertation,* University of North Carolina at Chapel Hill, 1999.

[Ben01] Bennett, D. Elumens. *Retrieved from the World Wide Web*: http://www.virtual-reality.com, 2001.

[Bie93] Bier, E., Stone, M., Pier, K., Buxton, W. and DeRose, T. Toolglass and Magic Lenses: The See-through Interface. *Computer Graphics (proceedings of the ACM SIGGRAPH '93)*, pp. 73-80, 1993.

[Bim99a] Bimber, O. Continuous 6D Gesture Recognition: A Fuzzy-Logic Approach. *In proceedings of 7-th International Conference in Central Europe on Computer Graphics, Visualization and Interactive Digital Media (WSCG'99)*, vol. 1, pp. 24-30, 1999.

[Bim99b] Bimber, O. Gesture Controlled Object Interaction: A Virtual Table Case-Study. *In proceedings of 7-th International Conference in Central Europe on Computer Graphics, Visualization and Interactive Digital Media (WSCG'99)*, vol. 1, pp. 17-23, 1999.

[Bim99c] Bimber, O. Rudiments of a 3D freehand sketch based human-computer interface for immersive virtual environments. *In proceedings of Virtual Reality Systems and Technology (VRST'99)*, vol. 1, pp. 182-183, 1999.

[Bim00a] Bimber, O., Encarnação, L.M., and Schmalstieg, D. Real Mirrors Reflecting Virtual Worlds. *In proceedings of IEEE Virtual Reality (IEEE VR'00)*, pp. 21-28, 2000.

[Bim00b] Bimber, O., Encarnação, L.M., and Schmalstieg, D. Augmented Reality with Back-Projection Systems using Transflective Surfaces. *Computer Graphics Forum (proceedings of EUROGRAPHICS 2000 - EG'2000)*, vol. 19, no. 3, pp.161-168, 2000.

[Bim00c] Bimber, O., Encarnação, L.M. and Stork, A. A Multi-Layered Architecture for Sketch-based Interaction within Three-dimensional Virtual Environments. *Computers and Graphics - The international Journal of Systems and Applications in Computer Graphics*, vol. 24, no. 6, pp. 851-867, 2000.

[Bim01a] Bimber, O., Encarnação, L.M. and Stork, A. Seamless integration of virtual reality in habitual workplaces. *Journal for Industrial Science, Munich University of Technology*, vol.55, no.2., pp. 103-112, 2001.

[Bim01b] Bimber, O., Stork, A., and Branco, P. Projection-based Augmented Engineering. *Human-Computer Interaction International (HCII'01)*, vol. 1, pp. 787-791, 2001.

[Bim01c] Bimber, O., Fröhlich, B., Schmalstieg, D., and Encarnação, L.M. Virtual Showcases - Presenting Hybride Exhibits. *SIGGRAPH'01 sketches and applications*, pp. 277, 2001.

[Bim01d] Bimber, O., Fröhlich, B., Schmalstieg, D., and Encarnação, L.M. The Virtual Showcase. *IEEE Computer Graphics & Applications*, vol. 21, no.6, pp. 48-55, 2001.

[Bim01e] Bimber, O., Encarnação, L.M. and Branco, P. The Extended Virtual Table: An Optical Extension for Table-Like Projection Systems. *Presence: Teleoperators and Virtual Environments*, vol.10, no. 6, pp. 613-631, 2001.

[Bim02a] Bimber, O. and Fröhlich, B. Occlusion Shadows: Using Projected Light to Generate Realistic Occlusion Effects for View-Dependent Optical See-Through Displays. *In proceedings of International Symposium on Mixed and Augmented Reality (ISMAR'02)*, 2002.

[Bim02b] Bimber, O., Gatesy, S.M., Witmer, L.M., Raskar, R. and Encarnação, L.M. Merging Fossil Specimens with Computer-Generated Information. *IEEE Computer*, September, pp. 45-50, 2002.

[Bli76] Blinn, J. F. and Newell, M. E. Texture and reflection in computer generated images. *Communications of the ACM*, vol. 19, pp. 542-546, 1976.

[Bli88] Blinn, J.F. Me and my (fake) shadow. *IEEE Computer Graphics and Applications*, vol. 8, no. 1, pp. 82-86, 1988.

[Blink95] Blinkenstrofer, C.H. Graffiti. *Pen Computing*, pp. 30-31, 1995.

[Blu00] Blundell, B. and Schwarz, A. Volumetric Three-dimmensional Display Systems. *New York: Wiley-Interscience*, 2000.

[Bre96] Breen, D.E., Whitaker, R. T., Rose, E. and Tuceryan, M. Interactive Occlusion and Automatic Object Placement for Augmented Reality. *Computer and Graphics Forum (proceedings of EUROGRAPHICS'96)*, vol. 15, no. 3, pp. C11-C22, 1996.

[Bren73] Brent, R.P. Algorithms for minimization without derivatives. *Prentice-Hall, Engelwood Cliffs*, NJ, 1973.

[Bry97] Bryson, S., Zeltzer, D., Bolas, M.T., de La Chapelle, and Bennett, D. The Future of Virtual Reality: Head-Mounted Displays Versus Spatially Immersive Displays. *Computer Graphics (proceedings of SIGGRAPH'97)*, pp. 485-486, 1997.

[Bur90] Burroughes, J.H. Light Emitting Polymers. *Nature*, pp. 347, 1990.

[Chi95] Chinnock, C. Holographic 3-D images float in free space. *Laser Focus World*, vol. 31, no. 6, pp. 22-24, 1995.

[Coq99] Coquillart, S., and Wesche, G. The Virtual Palette and the Virtual Remote Control Panel: A Device and an Interaction paradigm for the Responsive Workbench. *In proceedings of the 1999 IEEE Virtual Reality Conference*, pp. 213-216, 1999.

[Cru93] Cruz-Neira, C., Sandin, D., and DeFanti, T. Surround-Screen Projection-Based Virtual Reality: The Design and Implementation of the CAVE. *Computer Graph ics (proceedings of SIGGRAPH'93)*, pp. 135-142, 1993.

[Die96] Diefenbach, P. J. Pipeline Rendering: Interaction and Realism through Hardware-Based Multi-Pass Rendering. *Ph.D. Dissertation*, University of Pennsylvania (USA), 1996.

[Die97] Diefenbach, P. J. and Badler, N. I. Multi-pass pipeline rendering: realism for dynamic environments. *In proceedings of Symposium on Interactive 3D Graphics'97, ACM Press*, 1997.

[Dim01] Dimensional Media Associates, Inc., *Retrieved from the World Wide Web:* http://www.3dmedia.com/, 2001.

[Dow96] Downing, E.A., Hesselink, L., Ralston, J. and Macfarlane, R. A three-color, solid-state three-dimensional display. *Science*, vol. 273, pp. 1185-1189, 1996.

[Dun01] Dunkane, Corp. Mimio, *Retrieved from the World Wide Web*: http://www.dukane.com/AudioVisual/whiteboard.htm, 2001.

[Eck98] Eckel, G. OpenGL Volumizer Programmer's Guide. Silicon Graphics Inc., *Retrieved from the World WideWeb:* http://www.sgi.com/software/volumizer/tech_info.html, 1998.

[Edm01] Edmund Industrial Optics. Edmund Scientific. *Retrieved from the World Wide Web:* http://www.edmundscientific.com/optics, 2001.

[Eli72] Elings, V. B. and Landry, C. J. Optical display device. *U.S. Patent*, No. 3,647,284, 1972.

[Enc99] Encarnação, L.M., Bimber, O., Schmalstieg, D., and Chandler, S.D. A Translucent Sketchpad for the Virtual Table Exploring Motion-based Gesture Recognition. *Computer Graphics Forum (proceedings of EUROGRAPHICS'99)*, vol. 19, no. 3, pp. C-277 - C-285, NCC Blackwell, 1999.

[Enc00a] L.M. Encarnação, O. Bimber, D. Schmalstieg, and R. J. Barton III. Seamless 3D interaction for Virtual Tables, projection planes, and CAVEs. *In Cockpit Displays VII: Displays for Defense Applications, Darrel G. Hopper, Editor (proceedings of SPIE)*, vol. 4022, pp. 177-188, 2000.

[Enc00b] Encarnação, L.M., Bimber, O., Schmalstieg, D. and Barton III, R.J. Walk-up VR: Virtual Reality beyond Projection Screens. *IEEE ComputerGraphics and Applications*, vol. 20, no. 6, pp. 19-23, 2000.

[Fak01a] Fakespace Labs, Inc. BOOM 3C. *Retrieved from the World Wide Web:* http://www.fakespacelabs.com/products/boom3c.html, 2001.

[Fak01b] Fakespace Systems, Inc. ImmersaDesk Series. *Retrieved from the World Wide Web:* http://www.fakespacesystems.com/products.html, 2001.

[Fav99] Favalora, G.E. Multiplanar autostereoscopic imaging system. *U.S. Patent,* No. 5,936,767, 1999.

[Fav00] Favalora, G.E., Hall, D.M., Giovinco, M., Napoli, J., and Dorval, R.K. A Multi-Megavoxel Volumetric 3-D Display System for Distributed Collaboration. *In proceedings of IEEE GlobeCom 2000 Workshop,* Applications of Virtual Reality Technologies for future Telecommunication Systems, San Francisco, 2000.

[Fei93] Feiner, S., MacIntyre, B., et al. Windows on the World: 2D Windows for 3D Augmented Reality. *In proceedings of ACM Symposium on User Interface Software and Technology*, pp. 145-155, 1993.

[Fit93] Fitzmaurice, G.W. Situated Information Spaces and Spatially Aware Palmtop Computer. *In proceedings of CACM'93,* vol. 35, no. 7, pp. 38-49, 1993.

[Fox98] Foxlin, E., Harrington, M., and Pfeifer, G. Constellation: A Wide-Range Wireless Motion-Tracking System for Augmented Reality and Virtual Set Applications. *In prooceedings of SIGGRAPH'98,* pp. 371-378, 1998.

[Fro00] Fröhlich, B., & Plate, J. The Cubic Mouse: A New Device for Three-Dimensional Input. *In proceedings of ACM CHI 2000,* pp. 526-531, 2000.

[Fuc82] Fuchs, H., Pizer, S. M., Tsai, L. C., and Bloombreg, S. H. Adding a True 3-D Display to a Raster Graphics System. *IEEE Computer Graphics and Applications*, vol. 2, no. 7, pp. 73-78, 1982.

[Fuh99a] Fuhrmann, A., Hesina, G., Faure, F., and Gervautz, M. Occlusion in Collaborative Augmented Environments. *In proceedings of 5th Eurographics Workshop on Virtual Environments*, pp. 179-190, 1999.

[Fuh99b] Fuhrmann, A., Schmalstieg, D., and Purgathofer, W. Fast Calibration for Augmented Reality. *In proceedings of ACM Symposium on Virtual Reality Software and Technology 1999 (VRST'99),* pp. 166-167, 1999.

[Fuj86] Fujimoto, A., Tanaka, T. and Iwata, K. ARTS: Accelerated ray-tracing system. *IEEE Computer Graphics and Applications*. vol. 6, no. 4, pp. 16-26, 1986.

[Fun93] Funkhouser, T. and Sequin, C. Adaptive display algorithm for interactive frame rates during visualization of complex virtual environments. *Computer Graphics (proceedings of SIGGRAPH'93),* pp. 247-254, 1993.

[Fun00] Funkhouser, T. and Kai, L. (editors). Large Format Displays. *Computer Graphics and Applications,* Special Issue on Large Displays, vol. 20, no. 4, 2000.

[Gla84] Glassner, A. S. Space subdivision for fast ray tracing. *IEEE Computers Graphics and Applications*. vol. 4, no. 10, pp. 15-22, 1984.

[Gla89] Glassner, A. S. An Introduction to ray-tracing. *Academic Press*, 1989.

[Gre86] Greene, N. Environemnt mapping and other applications of world projections. *IEEE Computer Graphics and Applications*, vol. 6, no. 11, 1983.

[Gro95] Gross, M., Gatti, R., and Staadt, O. Fast multiresolution surface meshing. *In proceedings of IEEE Visualization'95*, pp. 135-142, 1995.

[Grot96] Gortler, S. J., Grzeszczuk, R., Szeliski R., and Cohen, M. F. The Lumigraph. *Computer Graphics (proceedings of SIGGRAPH'96)*, pp. 43-54, 1996.

[Hab73] Haber, R. N. and Hershenson, M. The psychology of visual perception. *New York: Holt*, Rinehart and Winston (publishers), 1973.

[Hae93] Haeberli, P. and Segal, M. Texture mapping as a fundamental drawing primitive. *In proceedings of 4th EUROGRAPHICS Workshop on Rendering*, pp. 259-266, 1993.

[Hal83] Hall, R.A. and Greenberg, D.P. A test for realistic image syntehsis. *IEEE Com puter Graphics and Applications*. vol. 2, no. 11, pp. 10-20, 1983.

[Hall97] Halle, M. Autostereoscopic displays and computer graphics. *Computer Graphics (proceedings of SIGGRAPH'97)*, vol. 31, no. 2, pp. 58-62, 1997.

[Hec84] Heckbert, P. and Hanrahan, P. Beam tracing polygonal objects. *Computer Graphics (proceedings of SIGGRAPH'84)*, vol. 18, no. 3, pp. 119-127, 1984.

[Hec96] Heckbert, P., Herf, M. Fast Soft Shadows. *Visual Proceedings of SIGGRAPH'96*, pp. 145, 1996.

[Hec97] Heckbert, P., Herf, M. Simulating Soft Shadows with Graphics Hardware. *Technical Report*, Carnegie Mellon University, no. CMU-CS-97-104, 1997.

[Hei98] Heidrich, W. and Seidel, H.P. View-independent environemnt maps. *In proceedings of EUROGRAPHICS Workshop on Graphics Hardware*, pp. 39-45, 1998.

[Hei99] Heidrich, W. and Seidel, H.P. Realistic, hardware-accelerated shading and lighting. *Computer Graphics (proceedings of SIGGRAPH'99)*, 1999.

[Hei00] Heidrich, W. Interactive Display of Global Illumination Solutions for Non-Diffuse Environments. *State of The Art Report EUROGRAPHICS'00*, State of the Art Reports, pp. 1-19, 2000.

[Hir97] Hirose, M., Ogi, T., Ishiwata, S., and Yamada, T. A Development of Immersive Multiscreen Displays (CABIN). *In proceedings of the VRSJ 2nd conference*, pp. 137-140, 1997.

[Hop96] Hoppe, H. Progressive meshes. *Computer Graphics (proceedings of SIGGRAPH'96)*, pp. 99-108, 1996.

[Hop97] Hoppe, H. View-dependent refinement of progressive meshes. *Computer Graphics (proceedings of SIGGRAPH'97)*, pp. 189-197, 1997.

[Hop98] Hoppe, H. Smooth view-dependent level-of-detail control and its application to terrain rendering. *In proceedings of IEEE Visualization'98*, pp. 35-42, 1998.

[How01] Howard, W. E. Organic Light Emitting Diodes (OLED). *OLED Technology Primer, Retrieved from the World Wide Web:* http://www.emagin.com/oledpri.htm, 2001.

[Ibm01] IBM Corp. ViaVoice. *Retrieved from the World Wide Web:* http://www-4.ibm.com/software/speech/, 2001.

[Ina00] Inami, M., Kawakami, N., Sekiguchi, D., Yanagida, Y., Maeda, T. and Tachi, S. Visuo-haptic display using head-mounted projector, *In proceedings IEEE Virtual Reality'00*, pp.233-240, 2000.

[Jan93] Janin, A., Mizell, D. and Caudell, T. Calibration of head-mounted displays for augmented reality applications. *In proceedings of the Virtual Reality Annual International Symposium (VRAIS'93)*, pp. 246-255, 1993.

[Joh99] Johnson, D. M., & Steward, J.E. (1999). Use of Virtual Environments for the Acquisition of Spatial Knowledge: Comparison Among Different Visual Displays. *Military Psychology*, vol. 11, no.2, pp. 129-148.

[Kai01] Kaiser Electro-optics, Inc. ProView SIMEYE XL100. *Retrieved from the World Wide Web:* http://www.cts.com/browse/keo/index.html, 2001.

[Kat99] Kato, H., Billinghurst, M., Blanding, R., and May. R. The ARToolkit. *Retrieved from the World Wide Web:* http://www.hitl.washington.edu/research/shared_space, 1999.

[Kal93] Kalawsky, R.S. The science of virtual reality and virtual environments. *Adisson-Wesley*, 1993.

[Kij97] Kijima, R. and Ojika, T. Transition between virtual environment and workstation environment with projective head-mounted display, *In proceedings of IEEE Virtual Reality Annual International Symposium*, pp.130-137, 1997.

[Kit01] Kitamura, Y., Konishi, T., Yamamoto, S., Kishino, F. Interactive Stereoscopic Display for Three or More Users. *Computer Graphics (proceedings of SIGGRAPH'01)*, 2001, pp. 231-239.

[Kiy00] Kiyokawa, K., Kurata, Y. and Ohno, H. An Optical See-through Display for Mutual Occlusion of Real and Virtual Environments. *In proceedings of IEEE & ACM ISAR 2000, pp. 60-67, 2000.*

[Kno77] Knowlton, K. C. Computer Displays Optically Superimpose on Input Devices. *Bell Systems Technical Journal*, vol. 53, no. 3, pp. 36-383, 1977.

[Knu85] Knuth, D. E. Dynamic Huffman coding. *Journal of Algorithms*, vol. 6, no. 2, pp.163-180, 1985.

[Kol93] Kollin, J. A Retinal Display For Virtual-Environment Applications. *In proceedings of SID International Symposium, Digest Of Technical Papers*, pp. 827, 1993.

[Kor91] Korsch, D. Reflective Optics. *Academic Press*, ISBN 0-12-421170-4, 1991.

[Kre92] Krech, D., Crutchfield, R.S. Grundlagen der Psychologie, *Belz Psychologie Verlags Union*, ISBN 3-621-27144-9, 1992.

[Kru94] Krueger, W., and Fröhlich, B. The responsive workbench. *IEEE Computer Graphics and Applications,* pp. 12-15, 1994.

[Kru95] Krueger, W., Bohn, C. A., Fröhlich, B., Schüth, H., Strauss, W., and Wesche, G. The responsive workbench: A virtual work environment. *IEEE Computer*, vol. 28, no. 7, pp. 42-48, 1995.

[Lec00] Lécuyer, A., Coquillart, S., Kheddar, A., Richard, P., & Coiffet, P. (2000). Pseudo-Haptic Forcefeedback: Can Isometric Input Devices Simulate Force Feedback?. *In proceedings of IEEE VR'00*, pp. 83-90, 2000.

[Lev96] Levoy, M. and Hanraham, P. Light field rendering. *Computer Graphics (proceed ings of SIGGRAPH'96)*, pp. 31-42, 1996.

[Lin01] Lindholm, E., Kilgard, M. J., Moreton, H. A User-Programmable Vertex Engine. *Computer Graphics (proceedings of SIGGRAPH'01)*, pp. 149-158, 2001.

[Linds96] Lindstrom, P., Koller, D., Ribarsky, W., Hughes, L., Faust, N., and Turner, G. Realtime, continuous level of detail rendering for height fields. *Computer Graphics (proceedings of SIGGRAPH'96),* pp.109-118, 1996.

[Loe96] Löffelmann, H., Gröller, E. Ray Tracing with Extended Cameras. *Journal of Visualization and Computer Animation*, vol. 7, no. 4, pp. 211-228, 1996.

[Luc97] Lucente, M. Interactive three-dimensional holographic displays: seeing the future in depth. *SIGGRAPH Computer Graphics*, special issue on “Current, New, and Emerging Display Systems“, 1997.

[Mas94] Massie, T. H., and Salisbury, J. K. The PHANTOM Haptic Interface: A Device for Probing Virtual Objects. *In proceedings of the ASME Dynamic Systems and Control Division*, 55(1), 295-301, 1994.

[Mck99a] McKay, S., Mason, S., Mair, L. S., Waddell, P., and Fraser, M. Membrane Mirror Based Display For Viewing 2D and 3D Images. *In proceedings of SPIE*, vol. 3634, pp. 144-155, 1999.

[Mck99b] McKay, S., Mason, S., Mair, L. S., Waddell, P., and Fraser, M. Stereoscopic Dis play using a 1.2-M Diameter Stretchable Membrane Mirror. *In proceedings of SPIE*, vol. 3639, pp. 122-131, 1999.

[Med01] Medical Imaging Group. The Dextroscope. *Retrieved from the World Wide Web:* http://www.krdl.org.sg/RND/biomed/WWW_Dextroscope, 2001.

[Mel96] Melzer, J. and Moffitt, K. Head-Mounted Displays: Designing for the User. Optical and Electro-Optical Engineering Series, Fisher and Smith (eds.), McGraw Hill, ISBN 0070418195, 1996.

[Mil94a] Milgram, P. and Kishino, F. A Taxonomy of Mixed Reality Visual Dis-plays. *IEICE Transactions on Information Systems E77-D*, vol.12, pp. 1321-1329, 1994.

[Mil94b] Milgram, P., Takemura, H., et al. Augmented Reality: A Class of Displays on the Reality-Virutality Continuum. *In proceedings of SPIE: Telemanipulator and Telepresence Technologies,* H. Das, SPIE. 2351, pp. 282-292, 1994

[Mit92] Mitchell, D. and Hanrahan, P. Illumination from Curved Reflectors. *Computer Graphics (proceedings of SIGGRAPH'92)*, pp. 283-291, 1992.

[Miz88] Mizuno, G. Display device. *U.S. Patent,* No. 4,776,118, 1988.

[Moe97] Möller, T., and Trumbore, B. Fast, Minimum Storage Ray-Triangle Intersection. *Journal of graphics tools*. vol. 2, no. 1, pp. 21-28, 1997.

[Nei93] Neider, J., Davis, T., and Woo, M. OpenGL programming Guide. *Addison-Wesley Publ.*, ISBN 0-201-63274-8, 1993.

[Nuv01] NuVision3D Technologies, Inc. 60GX. *Retrieved from the World Wide Web:* http://www.nuvision3d.com/the60gx.html, 2001.

[Nvid01a] Nvidia, Corp. GeForce 3. *Retrieved from the World Wide Web:* http://www.nvidia.com, 2001.

[Nvid01b] Nvidia, Corp. Unofficial NVIDIA stereoscopic driver support page. *Retrieved from the World Wide Web:* http://www.stereovision.net/articles/nvidiastereo/nvidiastereo.htm, 2001.

[Nvi01] N-Vision, Inc., Datavisor, *Retrieved from the World Wide Web:* http://www.nvis.com, 2001.

[Ofe98] Ofek, E. and Rappoport A. Interactive reflections on curved objects. *Computer Graphics (proceedings of SIGGRAPH'98)*, pp. 333-342, 1998.

[Ofe99] Ofek. E. Interactive Rendering of View-Dependent Global Lighting Phenomena. *Ph.D. Dissertation*, Hebrew University (Israel), 1999.

[Ogi01] Ogi, T., Yamada, T., Yamamoto, K. and Hirose, M. Invisible Interface for Immersive Virtual World. *In proceedings of the Immersive Projection Technology Workshop (IPT'01)*, pp. 237-246, Stuttgart, Germany, 2001.

[Par65] Parks, T.E. Post Retinal Visual Storage. *American Journal of Psychology*, vol. 78, pp. 145-147, 1965.

[Pars98] Parsons, J., and Rolland, J.P., A non-intrusive display technique for providing real-time data within a surgeons critical area of interest, *In proceedings of Medicine Meets Virtual Reality'98*, pp. 246-251, 1998.

[Pat92] Patterson, R. Human Stereopsis. *Human Factors*, vol. 34, no. 2, pp. 669-692, 1992.

[Patr00] Patrick, E., Cosgrove, D., Slavkovic, A., Rode, J.A., Verratti, T., and Chiselko, G. Using a Large Projection Screen as an Alternative to Head-Mounted Displays for Virtual Environments. *In proceedings of CHI' 2000*, vol. 2, no. 1, pp. 479-485, 2000.

[Per96] Pérez, J-P. Optik. *Spektrum, Akademischer Verlag*, ISBN 3-86025-389-1, 1996.

[Perl00] Perlin, K., Paxia, S., and Kollin, J.S. An autostereoscopic display. *Computer Graphics (proceedings of SIGGRAPH'00)*, pp. 319-326, 2000.

[Pol99] van de Pol, R., Ribarsky, W., Hodges, L. Interaction Techniques on the Virtual Workbench. *In proceedings of Virtual Environments'99*, pp. 157-168, 1999.

[Pop96] Poupyrev, I., Billinghurst, M., Weghorst, S., and Ichikawa, T. (1996). Go-Go Interaction Technique: Non-linear Mapping for Direct Manipulation in VR. *In proceedings of UIST'96*, pp. 79-80, 1996.

[Pos94] Poston, T. and Serra, L. The Virtual Workbench: Dextrous VR. *In proceedings of Virtual Reality Software an Technology (VRST'94)*, pp. 111-121, 1994.

[Pre92] Press, W.H., Teukolsky, S.A., Vetterling, W.T. and Flannery, B.P. Numerical Recipes in C - The Art of Scientific Computing (2nd edition), *Cambridge University Press, ISBN 0-521-43108-5*, pp. 412-420,1992.

[Pro01] G+B Pronova GmbH. HoloPro. *Retrieved from the World Wide Web:* http://www.holopro.html, 2001.

[Pry98] Pryor, Homer L., Furness, Thomas A. and Viirre, E. The Virtual Retinal Display: A New Display Technology Using Scanned Laser Light. *In proceedings of Human Factors and Ergonomics Society*, 42nd Annual Meeting, pp. 1570-1574, 1998.

[Ras98a] Raskar, R., Welch, G., Cutts, M., Lake, A., Stesin, L. and Fuchs, H. The Office of the Future : A Unified Approach to Image-Based Modeling and Spatially Immersive Displays. *Computer Graphics (proceedings of SIGGRAPH'98)*, pp. 179-188, 1998.

[Ras98b] Raskar, R., Cutts, M., Welch, G., and Stuerzlinger, W. Efficient Image Generation for Multiprojector and Multisurface Displays. *In proceedings of 9th Eurographics Workshop on Rendering*, pp. 139-144, 1998.

[Ras98c] Raskar, R., Welch, G., and Fuchs, H. Spatially Augmented Reality. *In proceedings of First IEEE Workshop on Augmented Reality (IWAR'98)*, San Francisco, CA, pp. 63-72, 1998.

[Ras99] Raskar, R., Welch, G., and Chen, W-C. Table-Top Spatially Augmented Reality: Bringing Physical Models to Life with Projected Imagery. *In proceedings of Second International IEEE Workshop on Augmented Reality (IWAR'99)*, San Francisco, CA, pp. 64-71, 1999.

[Ras01] Raskar, R. Welch, G., Low, K.L., and Bandyopadhyay, D. Shader Lamps: Animating real objects with image-based illumination. *In Proceedings of Eurographics Rendering Workshop (EGRW'01)*, 2001.

[Rea01] Reach-In Technologies. The Reach-In Display. *Retrieved from the World Wide Web:* http://www.reachin.se, 2001.

[Rol93] Rolland, J.P., and Hopkins, T. A Method of Computational Correction for Optical Distortion in Head-Mounted Displays. (Tech. Rep. No. TR93-045). UNC Chapel Hill, Department of Computer Science, 1993.

[Rol94] Rolland, J., Rich, H. and Fuchs, H. A Comparison of Optical and Video See-Through Head-Mounted Displays. *In proceedings of SPIE: Telemanipulator and Telepresence Technologies*, pp. 293-307, 1994.

[Rub80] Rubin, S. and Whitted, T. A three-dimensional representation for fast rendering of complex scenes. *Computer Graphics*, vol. 14, no. 3, pp. 110-116, 1980.

[Rus00] Rusinkiewicz, S. and Levoy, M. QSplat: A Multiresolution Point Rendering System for Large Meshes. *Computer Graphics (proceedings of SIGGRAPH'00)*, pp. 343-352, 2000.

[Schm99] Schmalstieg, D., Encarnação, L.M., Szalavári, Zs. Using Transparent Props for Interaction with the Virtual Table. *In proceedings of ACM SIGGRAPH Symposium on Interactive 3D Graphics (I3DG'99)*, Atlanta, GA. ACM Press, pp. 147-153, 1999.

[Sch83] Schmandt, C. Spatial Input/Display Correspondence in a Stereoscopic Computer Graphics Workstation. *Computer Graphics (proceedings of SIGGRAPH'83)*, vol. 17, no. 3, pp. 253-261, 1983.

[Seg92] Segal, M., Korobkin, C., van Widenfelt, R. Foran, J., and Haeberli, P. Fast Shaddows and Lighting Effects Using Texture Mapping. *Computer Graphics (proceedings of SIGGRAPH'92)*, vol. 26, no. 2, pp. 249-252, 1992.

[Sil01] Silicon Graphics, Inc. Powerwall. *Retrieved from the World Wide Web:* http://www.stereographics.com/html/body_powerwall.html, 2001.

[Son00] Sony, Corp., Glasstron, *Retrieved from the World Wide Web:* http://www.ita.sel.sony.com/products/av/glasstron, 2000.

[Spa92] Spanguolo, M. Polyhedral surface decomposition based on curvature analysis. *Modern Geometric Computing for Visualization,* Springer Verlag, 1992.

[Stan98] Stanny, K.M., Mourant, R.R., and Kennedy, R.S. Human Factors Issues in Virtual Environments - A Review of Literature. *Presence: Teleoperators and Virtual Environments,* vol. 7. no. 4, pp. 327-351, 1998.

[Star83] Starkey D. and Morant, R.B. A technique for making realistic three-dimensional images of objects. *Behaviour Research Methods and Instrumentation*, vol. 15, no. 4, pp. 420-423, 1983.

[Ste01] StereoGraphics, Corp. ChrystalEyes. *Retrieved from the World Wide Web:* http://www.stereographics.com/html/body_crystaleyes.html, 2001.

[Stet01] Stetten, G., Chib, V., Hildebrand, D., Bursee, J. Real Time Tomographic Reflection: Phantoms for Calibration and Biopsy, *In proceedings of IEEE/ACM International Symposium on Augmented Reality (ISMAR'01),* pp. 11-19.

[Sto00] Stork, A. Effiziente 3D-Interaktions- und Visualisierungstechniken fuer benutzerzentrierte Modellierungssysteme. *Ph.D. Dissertation,* Darmstadt University of Technology, 2000.

[Stol01] Stoll, G., Eldridge, M., Patterson, D., Webb, A., Berman, S., Levy, R., Caywood, C., Taveira, M., Hunt, S., Hanrahan, P. Lightning-2: A high-performance subsystem for PC clusters. *Computer Graphics (proceedings of SIGGRAPH'01),* 2001, pp. 141-148.

[Sum94] Summer S. K., et. al. Device for the creation of three-dimensional images. *U.S. Patent,* No. 5,311,357, 1994.

[Sut65] Sutherland, I.E. The Ultimate Display. *In proceedings of IFIP'65,* 506-508, 1965.

[Sut68] Sutherland, I.E. A head mounted three dimensional display. *In proceedings of the fall joint computer conference (AFIPS)*. 33(1), 757-764, 1968.

[Sza97] Szalavári, Zs. and Gervautz., M. The Personal Interaction Panel - A Two Handed Interface for Augmented Reality. *Computer Graphics Forum (proceedings of EUROGRAPHICS'97),* vol. 16(3), pp. 335-346. NCC Blackwell, 1997.

[Tan01a] TAN Projectiontechnologies. Tanorama Cylinder. *Retrieved from the World Wide Web:* http://www.tan.de, 2001.

[Tan01b] TAN Projectiontechnologies. Responsive Workbench. *Retrieved from the World Wide Web:* http://www.tan.de, 2001.

[Tan01c] TAN Projectiontechnologies. Holobench. *Retrieved from the World Wide Web:* http://www.tan.de, 2001.

[Tay96] Taylor, D.C. and Barret, W. A. An Algorithm for continuous resolution polygonalizations of a discrete surface. *Computer Graphics (proceedings of SIGGRAPH'94)*, pp. 33-42, 1994.

[Tra67] Traub, A.C. Stereoscopic Display Using Varifocal Mirror Oscillations. *Applied Optics*, vol 6, no. 6, pp. 1085-1087, 1967.

[Vanb00] Van Belle R., Maximus, B., Vandenbogaerde, P. Clodfelter, R. High Quality Geometry Distortion Tool for Use with LCD and DLP Projectors. *In proceedings of IMAGE'00, pp. , 2000.*

[Vie96] Viega, J., Conway, M., Williams, G. and Pausch, R. 3D Magic Lenses. *In proceedings of ACM USIT'96*, pp. 51-58, ACM Press, 1996.

[Vil97] Villasenor, J. and Mangione-Smith, W. H. Configurable Computing. *Scientific America*, pp. 54-59, 1997.

[Vir01] Virtual I/O Display Systems, LLC., i-Glasses, *Retrieved from the World Wide Web:* http://www.vio.com/html/body_i-glasses.php3, 2001.

[Voo94] Voorhies, D. and Foran, J. Reflection Vector Shading Hardware. *Computer Graphics (proceedings of SIGGRAPH'94)*, pp. 163-166, 1994.

[Wal94] Walker, M. Ghostmasters: A Look Back at America's Midnight Spook Shows. *Cool Hand Publ.*, ISBN 1-56790-146-8, 1994.

[War93] Ware, C., Arthur, K., et al. Fish Tank Virtual Reality. *In proceedings of Inter CHI'93 Conference on Human Factors in Computing Systems*, pp. 37-42, 1993.

[Wat95] Watson, B., and Hodges, L. Using Texture Maps to Correct for Optical Distortion in Head-Mounted Displays. *In proceedings of IEEE VRAIS'95*, pp. 172-178, 1995.

[Wel89] Welck, S. A. Real image projection system with two curved reflectors of paraboloid of revolution shape having each vertex coincident with the focal point of the other. *U.S. Patent*, No. 4,802,750, 1989.

[Wes91] Westover, L. Splatting: A Feed-Forward Volume Rendering Algorithm. *Ph.D. Dissertation*, University of North Carolina at Chapel Hill, 1991.

[Whi95] Whitaker, R., Crampton, C., Breen, D., Tuceryan, M. and Rose, E. Object Calibration for Augmented Reality. *Computer Graphics Forum (proceedings of EUROGRAPHICS'95)*, vol. 14, no. 3, pp. 15-27, 1995.

[Whit80] Whitted, T. An improved illumination model for shaded display. *Communications of the ACM*, vol. 26, no. 6, pp. 343-349, 1980.

[Wie99] Wiegand, T. E., von Schloerb, D. W., and Sachtler, W. L. Virtual Workbench: Near-Field Virtual Environment System with Applications. *Presence: Teleoperators and Virtual Environments,* vol. 8, no. 5, pp. 492-519, 1999.

[Win01] Winnov. Videum VO. *Retrieved from the World Wide Web:* http://www.cts.com/browse/keo/index.html, 2001.

[Yan01] Yang, R., Gotz, D. Hensley, J., Towles, H. and Borwn, M.S. PixelFlex: A Reconfigurable Multi-Projector Display System. *In proceedings of IEEE Visualization'01,* pp. 167-174, 2001.

Curriculum Vitae

Oliver Bimber was born in Bad Marieneberg, Germany in 1973. He is currently a scientist at the Bauhaus University Weimar, Germany. From 2001 to 2002 Bimber worked as a senior researcher at the Fraunhofer Center for Research in Computer Graphics in Providence, RI/ USA, and from 1998 to 2001 he was a scientist at the Fraunhofer Institute for Computer Graphics in Rostock, Germany. In 1998 he received the degree of Dipl. Inform. (FH) in Scientific Computing from the University of Applied Science Giessen and a B.Sc. degree in Commercial Computing from the Dundalk Institute of Technology. Bimber initiated the Virtual Showcase project in Europe and the Augmented Paleontology project in the USA.

In his career, Bimber received several scientific achievement awards and is author of more than thirty technical papers and journal articles. He was guest editor of the Computer & Graphics special issue on "Mixed Realities - Beyond Conventions", and has served as session chair and review committee member for several international conferences.

Bimber also gave a number of guest lectures at recognized institutions. Among them were Brown University, Princeton University, the IBM T.J. Watson Research Center, the Daimler-Chrysler Virtual Reality Competence Center, and the Mitsubishi Electronic Research Lab (MERL). His research interests include display technologies, rendering and human-computer interaction for Mixed Realities. Bimber is member of IEEE, ACM and ACM Siggraph.