Alternative Augmented Reality Approaches: Concepts, Techniques, and Applications

Oliver Bimber[1] and Ramesh Raskar[2]

[1]Bauhaus University, Weimar, Germany
T +49-3643-583724, F +49-3643-583709, Email: oliver.bimber@medien.uni-weimar.de, WWW: http://www.uni-weimar.de/~bimber

[2] MERL - Mitsubishi Electric Research Lab, Cambridge, USA
T +1-617-621-7533, F +1-617-621-7550, Email: raskar@merl.com, WWW : http://www.merl.com/people/raskar/ , http://www.raskar.com

Abstract

In this tutorial we discuss application specific alternative augmented reality (AR) approaches –such as Virtual Showcases and Shader Lamps– that focus on overcoming some of the limitations linked to conventional AR displays. State-of-the-art concepts, details about hard- and software implementations, and current areas of application are presented. An extensive overview over different stereoscopic and auto-stereoscopic display techniques is given. This enables readers to identify parallels between Virtual Reality and augmented reality displays concepts, and stimulate them to think about alternative approaches for AR. Our focus is on interactive rendering techniques that support display concepts, such as projector-based or spatial optical see-through AR. Methods that create graphical augmentations with respect to the applied optical elements and display surfaces, and techniques for creating a high level of consistency between real and virtual environments will be discussed. Finally, our experiences made with such technologies within art, edutainment, research and industrial areas will be shared with the audience. By handing out construction drawings and algorithms, we aim at enabling participants to realize such systems on their own.

Categories and Subject Descriptors (according to ACM CSS): H.5.1 [Multimedia Information Systems]: Artificial, Augmented and Virtual Realities; I.3.1 [Hardware Architecture]: Three-dimensional Displays; I.3.2 [Computer Graphics]: Graphics Systems; I.3.3 [Computer Graphics]: Picture/Image Generation – Display Algorithms, Viewing Algorithms; I.3.7 [Computer Graphics]: Color, Shading, Shadowing, and Texture

1. Introduction and overview

Video see-through and optical see-through head-mounted displays have been the traditional output technologies for augmented reality (AR) applications for more than forty years. However, they suffer from several technological and ergonomic drawbacks which prevent them from being used effectively in all application areas.

We will discuss application specific alternative AR approaches that focus on overcoming some of these limitations. Existing state-of-the-art concepts, details about hard- and software implementations, and current areas of application will be presented. The tutorial aims at enabling the participants to realize such systems on their own.

Chapter 2 gives an extensive overview over different display techniques. It will enable readers to identify parallels between virtual reality and augmented reality display technology, and stimulate them to think about alternative display approaches for AR. First, a classification of stereoscopic displays is presented. Auto-stereoscopic displays are described next. Although auto-stereoscopic displays are not common for augmented reality today, we believe that they will play a major role in the future. Consequently the different types are discussed. Potential applications of auto-stereoscopic techniques to field of AR will be described at the end of the course notes. Goggle-bound stereoscopic displays, such as head-attached and spatial displays will be outlined next. This will mainly be done within a virtual reality context. To bridge the gap between auto-stereoscopic displays and optical see-through AR, different mirror displays are presented. Finally, we discuss traditional as well as alternative AR display approaches – ranging form head-mounted displays, over spatial displays to object-oriented displays.

Chapter 3 describes augmented reality techniques using non-trivial projections screens. Our emphasis is on projector-based augmented reality. In this context, we describe two scenarios, a concept of spatial augmentation and then a more specific method of changing surface appearance of real objects.

In chapter 4 we first present two different examples for spatial optical see-through displays: The Extended Virtual Table as a single user setup, and the Virtual Showcase that supports multiple users. These examples will serve as references for describing the following rendering and illumination techniques. Interactive rendering techniques that can be used to drive spatial optical see-through AR displays which consist of a combination of single or multiple beam-splitters (planar or curved) and screens (projection displays or monitors) are discussed next. Finally, the projector-based illumination concept is described, and examples are outlined how it can be used to create consistent illumination and occlusion effects.

Chapter 5 will discuss two current areas of application of projector-based augmented reality and spatial optical see-through displays: scientific visualization and digital storytelling within a museum context; and vehicle simulation, and computer aided engineering and design. Concrete setups and demonstrations are presented that have been displayed to a large audience during several public events. User feedback and reactions are presented and discussed.

In chapter 6 we summarize our tutorial and give an outlook to enabling technologies that might influence augmented reality technology in the future. The possibilities and limitations of technologies, such as video projectors, organic light emitting diodes, light emitting polymers, electronic paper, particular solid state volumetric and parallax display approaches, and holography will be outlined.

In appendix 1 we present details about how to build a monitor-based Virtual Showcase, and appendix 2 outlines source code for calibrating projectors and cameras.

2. Stereoscopic display approaches – an overview

This section will provide a broad classification of today's stereoscopic displays (cf. figure 2.1). Note that we do not claim to present a complete list of existing systems and their variations, but rather focus on the technology that is (or might become) relevant for our tutorial.

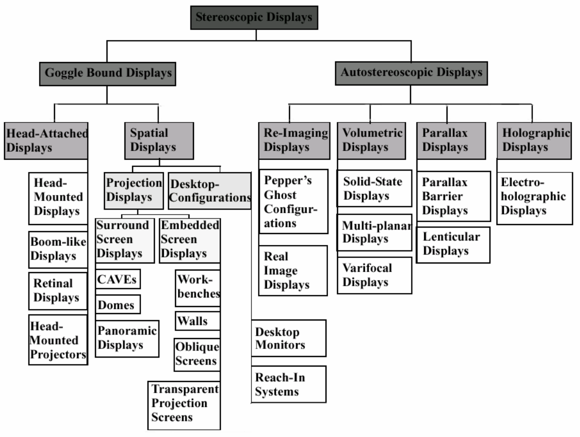


Figure 2.1: *Classification of stereoscopic displays.*

Stereoscopic displays can be divided into *autostereoscopic displays* and *goggle-bound displays*. While goggle-bound displays require the aid of additional glasses to support a proper separation of the stereo images, autostereoscopic displays do not.

While today the Virtual Realty community has oriented themselves manly towards projection displays, head-mounted displays are still the dominant display technology for almost all Augmented Realty applications.

We will start to discuss autostereoscopic displays in more detail in section 2.1 and describe goggle-bound displays in section 2.2.

2.1. Autostereoscopic displays

Autostereoscopic displays [43] present three-dimensional images to the observers without the need of additional glasses. Four classes of autostereoscopic displays can be found: *re-imaging displays, volumetric displays, parallax displays,* and *holographic displays*.

2.1.1. Re-imaging displays

Re-imaging displays project existing real objects to a new position or depth. They capture and re-radiate the light from the real object to a new location in space. An important characteristic of re-imaging displays is that they do not generate three-dimensional images by themselves. Some re-imaging systems use lenses and/or mirrors to generate copies of existing objects. Especially half-silvered mirror setups are used by theme parks to generate a copy of a real three-dimensional environment and overlay it over another real environment. These types of mirror displays - so called *Pepper's ghost configurations* [125] - generate virtual images and are further discussed below.

Other re-imaging displays apply more complex optics and additional display devices. For instance, some re-imaging displays generate a copy of a two-dimensional CRT screen which then appears to float in front of the optics. These types of mirror displays -so-called *real image displays* [31, 111, 79, 127, 114, 24, 73, 74, 28] - generate real images and are also discussed in more detail below.

Another example of a re-imaging display was used by Sega in an arcade video game to relay and distort the appearance of a flat CRT screen into a curved surface [43].

Re-imaging displays can be characterized by the following properties:

- Do not generate three-dimensional images by themselves;
- Generate visual copies of real objects (whereby the real object can be a computer-controlled screen);
- If a screen is re-displayed, the copy of the image that is shown on the screen remains two-dimensional (i.e., no autostereoscopic viewing is provided with respect to the displayed image);
- If a real object is re-displayed, the copy addresses the same visual depth cues as the original object (i.e., stereopsis, accommodation, vergence, parallax, etc.);
- With respect to re-displayed real objects, multiple observers are simultaneously supported.

Re-imaging displays are frequently applied as "eye-catchers" for product presentation by the advertising industry, or to facilitate special on-stage effects by the entertainment industry.

2.1.2. Volumetric displays

Volumetric displays [18] directly illuminate spatial points within a display volume. In contrast to re-imaging displays, volumetric displays can generate synthetic images of voxelized data or three-dimensional primitives. These types of displays generate images by filling or sweeping out a volumetric image space.

Solid-state devices are variations of volumetric displays which display voxel data within a translucent substrate by generating light points with an external source (for example with lasers of different wavelengths located outside the substrate that are scanned through the image space) [30]. *Multi-planar volumetric displays* build volumetric images from a time-multiplexed series of two-dimensional images. These images are displayed with a swiftly moving or spinning display element. This display element can be, for example, a rotating proprietary screen onto which the images are projected (e.g., using an external projector [36] or lasers [35]). Other systems directly move or spin light generating elements (e.g., light diodes). In either case, the human visual system interprets these time-multiplexed image slices as a three-dimensional whole.

Varifocal mirror displays [122, 39, 73, 74] are yet another group of volumetric displays. They apply flexible mirrors to sweep an image of a CRT screen through different depth planes of the image volume. These types of mirror displays are also discussed in more detail within the subsequent sections.

Regardless of the underlying technology, volumetric displays share the following characteristics [43]:

- Presented volume can be perceived from a wide range of viewpoints, surrounding the display;
- Simultaneous support of multiple observers;
- Sense of ocular accommodation is supported;
- Spatial resolution of the presented graphics is limited;
- View-dependent shading and culling (required to simulate occlusion) of the presented graphics is not supported.

Particularly due to the last point, volumetric displays are mainly applied to present wire-frame or icon-based contents.

2.1.3. Parallax displays

Parallax displays are display screens (e.g., CRT or LCD displays) that are overlaid with an array of light-directing elements [43]. Depending on the observer's location, the emitted light that is presented by the display is directed so that it appears to originate from different parts of the display while changing the viewpoint. If the light is directed to both eyes individually, the observer's visual system interprets the different light information to be emitted by the same spatial point.

Examples of parallax displays are *parallax barrier displays* that apply a controllable array of light-blocking elements (e.g., a light blocking film or liquid crystal barriers [89]) in front of a CRT screen. Depending on the observer's viewpoint, these light-blocking elements are used to direct the displayed stereo-images to the corresponding eyes.

Other examples are *lenticular sheet displays* that apply an array of optical elements (e.g., small cylindrical or spherical lenses) to direct the light for a limited number of defined viewing-zones.

Several properties characterize parallax displays [43]:

- Simulation of occlusion is supported;
- Sense of ocular accommodation is supported (if no information reduction techniques are used that diminish or eliminate ocular accommodation);
- Limited viewing angle and discretized viewing zones;
- Single viewer devices (head-tracking is required if motion parallax has to be supported);
- Usually address only horizontal parallax (only lenticular sheet displays that apply spherical lenses support full parallax);
- Correct viewing distance has to be kept;
- Spatial resolution of the presented graphics is limited.

Parallax displays can be published and mass-produced in a wide range of sizes, and can be used to display photo-realistic images.

2.1.4. Holographic displays

Holographic displays record the light's wavefront information that is emitted by an object within so-called *interference fringes.* The interference fringes can, under certain circumstances (if they are correctly illuminated), act as a complex diffractive lens that reconstructs the recorded light information (i.e., its direction and intensity). *Electroholographic displays* (such as the one described by Lucente [69]) create these interference fringes electronically from a connected raster-engine.

Holographic displays share most of the properties of volumetric displays. The following characteristics, however, differ from volumetric displays:

- Low data bandwidth of high-quality holograms;
- Very low resolution of the presented holograms is supported (due to low bandwidth);
- Usually only horizontal parallax is supported (due to low bandwidth);
- Very limited in presenting shading and color information (due to low bandwidth);
- Restricted viewing angle.

Currently, holographic display technology is still far from producing high-quality three-dimensional images using affordable hardware [43].

2.2. Goggle-bound displays

Goggle-bound displays require to wear additional goggle-like devices in front of the eyes to support a proper separation of the stereo images. They can be divided into *head-attached displays* and *spatial displays*.

2.2.1. Head-attached displays

Head-attached displays mostly provide individual display elements for each eye and consequently can present both stereo images simultaneously. Examples for such elements are miniature CRT or LCD screens that are applied in most *head-mounted displays* [115, 116, 4, 76, 52] and *BOOM-like displays* [32]. *Retinal displays* [62, 93] utilize low-power lasers to scan modulated light directly onto the retina of the human eye, instead of providing screens in front of the eyes. This produces a much brighter and higher resolution image with a potentially wider field of view than a screen-based display. *Head-mounted projective displays* [87, 50] or *projective head-mounted displays* [56] are projection-based alternatives that employ head-mounted miniature projectors instead of miniature displays. Such devices tend to combine the advantages of large projection displays with those of head-mounted displays.

The following characteristics can be related to head-attached displays:

- Simultaneous support of multiple observers (wearing individual devices);
- Mobile applications possible;
- Lack in resolution that is due to limitations of the applied miniature displays or projectors;
- Limited field of view that is due to limitations of the applied optics. Note that head-mounted projective displays and projective head-mounted displays address this problem;
- Sense of ocular accommodation is not supported due to a constant image depth and the resulting fixed focal length (for head-mounted displays and BOOM-like devices), or due to the complete bypass of the ocular motor-system by scanning directly onto the retina (retinal displays);
- In case of head-mounted projective displays: The inconsistency of accommodation and convergence is decreased since spatial projection surfaces are utilized;
- In case of retinal displays: Monochrome (red) images are presented since small blue and green lasers do not yet exist;
- Imbalanced ratio between heavy optics (that results in cumbersome and uncomfortable devices) and ergonomic devices with a low image quality;

- Increased incidence of discomfort due to simulator sickness in case of head-attached image planes (especially during fast head movements) [88].

Head-attached displays (especially head-mounted displays) are currently the display devices that are mainly used for augmented reality applications.

2.2.2. Spatial displays

Spatial displays apply screens that are spatially aligned within the environment. Nevertheless, the users have to wear field-sequential (LCD shutter-glasses [84, 112]) or light-filtering (polarization or color filters) goggles to support a correct separation of the stereo images. The stereo separation technique for spatial displays is generally known as *shuttering*, since each of the two stereo images which are presented on the same screen(s) has to be made visible to only one eye (i.e., the other image has to be blocked respectively -by shutting the eye). Depending on the shuttering technology, the stereo images are either presented time sequentially (i.e., with field-sequential goggles) or simultaneously (i.e., with light-filtering goggles).

Spatial displays can be further divided into desktop configurations and projection displays. Using desktop monitors as a possible stereoscopic display is the traditional desktop-VR approach (also referred to as fish tank VR [126]). Since desktop monitors (i.e., only CRT screens, but not LCD screens) provide the refresh rate of 120Hz that is required for a time-sequential shuttering, LCD shutter glasses are mostly applied for stereo separation. Note that older applications also use color-filtering glasses (e.g., red-green or blue-red filters) to separate monochrome stereo images. Fish tank VR setups are classified as non-immersive, since in contrast to large screens the degree of immersion is low. Reach-in systems represent another type of desktop configurations that consist of an upside-down CRT screen which is reflected by a small horizontal mirror. They are discussed in more detail below.

Projection displays currently apply cathode ray tube (CRT), liquid crystal display (LCD) or digital light (DLP) projectors to beam the stereo images onto single or multiple, planar or curved display surfaces. Two types of projections exist: With *front-projection,* the projectors are located on the same side of the display surface as the observer. Thus, the observer might interfere with the projection frustum and cast shadow onto the display surface. With *rear-projection* (or *back-projection*), the projectors are located on the opposite site of the display surface to avoid this interference problem.

Projection displays that first transmit the images through polarized light filters before they are diffused by the display surface require polarized glasses (i.e., glasses with corresponding polarization filters in front of each eye) to separate the images respectively. This technique is known as *passive shuttering*. For passive shuttering, at least two projectors are necessary to beam both polarized stereo images simultaneously onto the display surface. Note that special display surfaces are required for passive systems. These surfaces have to be built from a metallic material, since every organic material would reverse or destroy the polarization direction of the light and consequently would make the image separation fail.

Projection displays that beam both stereo images sequentially onto the display surface require field-sequential shutter glasses to separate the stereo images. This technique is known as *active shuttering*. For active shuttering, only one projector is necessary since the images are projected time sequentially. However, as with desktop monitors, these projectors have to support the required refresh rate of 120Hz.

Note that projection screens can either be *opaque* or *transparent* - depending on their application. Transparent projection screens are further discussed below.

Depending on the number and the shape of the spatially aligned display surfaces, we can divide projection displays into *surround screen displays* and *embedded screen displays*.

Surround screen displays surround the observers with multiple planar (e.g., CAVEs [25], CABINs [45]) or single curved display surfaces (e.g., Domes [7] or panoramic displays [119]) to provide an *immersive* VR experience. Thus, the observers are completely encapsulated from the real environment. Usually, multiple projectors are used to cover the extensive range of the projection surface(s).

In contrast to surround screen displays, *embedded screen displays* integrate single, or a small number of display surfaces into the real environment. Thus, the users are not immersed into an exclusively virtual environment, but can interact with a *semi-immersive* virtual environment that is embedded within the surrounding real environment. Horizontal, *workbench-like* [63, 64, 5, 6, 33, 119, 120] or vertical *wall-like* [108] display screens are currently the most common embedded screen displays.

Krueger's Responsive Workbench [63, 64] is one of the pioneering *workbench-like projection systems*. The Responsive Workbench consists of a video projector that projects high-resolution stereoscopic images onto a mirror located under the table, which, in turn, reflects them in the direction of the table top (a ground glass screen). Analyzing the daily work situation of different types of computer users, Krueger et al chose a workbench-like system as an adaptation to the human living and working environment.

Based on the Responsive Workbench metaphor, a rich palette of similar rear-projection devices are available today that differ in size, mobility and applied projection technology. Among these systems are Wavefront's ActiveDesk, Barco's BARON [5], Fakespace's ImmersaDesk Series [33], and the Responsive Workbench [120] itself, which is sold by TAN Projectiontechnologies.

While all these systems are single-sided projection devices, a few two-sided (L-shaped) systems have been developed to offer a larger and (because of the normally limited projection area) less constrained viewing space. TAN's Holobench [121], for instance, is an extension of the Responsive Workbench, and Barco's Consul [6] has been developed based on the BARON Virtual Table.

Over the last years, an enormous variety of applications (concerning almost all VR areas) that involve table-like projection systems have been described.

Oblique screen displays represent a generalization of embedded screen displays, whereby special display surfaces are not integrated explicitly into the real environment. Rather, the real environment itself (i.e., the walls of a room, furniture, etc.) provides implicit display surfaces [94, 95]. To support single or multiple front projections onto a multi-plane or curved display surface, a three-pass rendering method is applied. During the first rendering pass, the desired image of the virtual environment is generated from the observers current viewpoint. Then the generated image is projected out from the user's point of view onto a registered virtual model of the display surface that is aligned with its real counterpart. For this, projective textures [106] are applied. During the second pass, this textured model is rendered from the projector's point of view and is finally beamed onto the real display surface. If multiple projectors are used, the second pass has to be repeated for each projector individually. The generated images have to be geometrically aligned and color and edge blended appropriately to realize a seamless transition between them. This is usually done during the third rendering pass [94].

Several general characteristics of projection displays can be found:

- High resolution (especially with tiled projection displays that apply multiple projectors [40]);
- Lower incidence of discomfort due to simulator sickness than head-attached displays because of the spatially aligned image planes (fast head movements are not critical) [88];
- Normally do not support multiple users;
- Passive shuttering, active shuttering in combination with rear-projection, and transparent projection screens require special display surfaces;
- Passive shuttering lacks from restricted head rotations* (due to the horizontal polarization direction of the light) and ghosting effects (due to the limited filtering capabilities of high contrast image portions);
- Active shuttering requires fast projectors which provide a high refresh rate;
- Sense of ocular accommodation is not supported due to a spatially constant image plane. However, compared to head-attached displays, accommodation is improved since the image depth, and consequently the focal length is not constant and changes with a moving observer;
- Displays are stationary (i.e., mobile applications are not supported);
- Semi-immersive displays suffer from *window violation* (the clipping of the graphics by the display surface's physical edges).

Although head-attached (especially head-mounted) displays have a long tradition within the VR community, stereoscopic projection displays are currently the dominant output technology for Virtual Reality applications. Bryson [20] sees various advantages of spatial displays over head-mounted displays.

2.2.3. Multi-user approaches

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

*Except advanced circular polarization filters are applied.

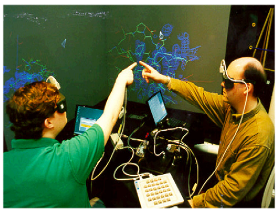


Figure 2.2: *Private screen example - UNC's Two-user Protein Interactive Theatre* [2]*. Courtesy: Arthur, Preston, Taylor, Brooks, Whitton, and Wright.*

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 [2] assigns two righted projection planes to individual observers. Thus, the two users' viewing-frustum cross each other while looking at the displays.

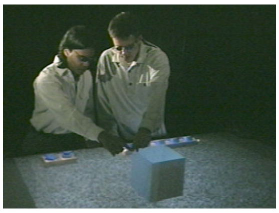


Figure 2.3: *Frame interleaving example - Stanford's Two-user Responsive Workbench* [1]*. Courtesy: Agrawala, Beers, Fröhlich, McDowall, Hanrahan, and Bolas.*

Stanford's two-user Responsive Workbench [1] is an example of 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.

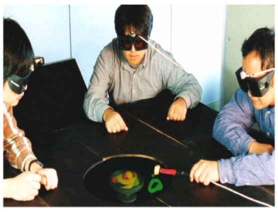


Figure 2.4: *screen partitioning example - Osaka University's IllusionHole* [57]*. Courtesy: Kitamura, Konishi, Yamamoto, Kishino.*

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 [57] applies a simple canopy that allows to perceive the individual image through a small hole, whereby the other images are covered by the canopy.

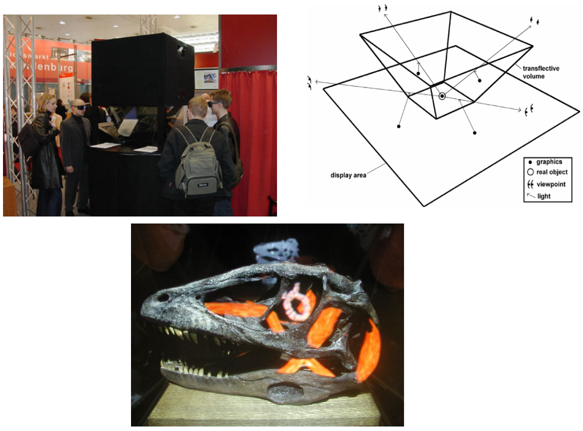


Figure 2.5: *Another screen partitioning example – The Virtual Showcase (pyramid-shaped variation)* [10]*.*

Virtual Showcases [10] (cf. figure 2.5 and 2.11) are also screen partitioning systems that use multi-mirror or curved mirror configurations to achieve an image separation.

2.3. Mirror displays

Beside several optical see-through head-mounted displays and head-mounted projector displays, a number of other display systems exist that apply full or half-silvered mirrors to achieve optical effects, such as an optical combination of graphics with the real environment. In this section, we want to discuss the different variations of *mirror displays*. Note that we describe only selected systems to introduce the corresponding display category they belong to, rather than presenting a complete list of systems.

2.3.1. Pepper's ghost configurations

Pepper's Ghost Configurations [125] are a common theatre illusion from around the turn of the century named after John Henry Pepper - a professor of chemistry at the London Polytechnic Institute. They belong to the class of re-imaging displays. At its simplest, a Pepper's ghost configuration consists of a large plate of glass that is mounted in front of a stage (usually with a 45° angle towards the audience). Looking through the glass plate, the audience is able to simultaneously see the stage area and, due to the self-reflection property of the glass, a mirrored image of an off-stage area below the glass plate. Different Pepper's ghost configurations are still used by entertainment and theme parks (such as the Haunted Mansion at Disney World) to present their special effects to the audience. Some of those systems reflect large projection screens that display prerecorded 2D videos or still images instead of real off-stage areas. The setup at London's Shakespeare Rose Theatre, for instance, applies a large 45° half-silvered mirror to reflect a rear-projection system that is aligned parallel to the floor.

Augmented reality applies stereoscopic displays that are extended by optical elements to overlay stereoscopic 3D graphics of a real environment. In an AR context, this requires a view-dependent rendering. The main drawback of a Pepper's ghost configuration is that the viewers' parallax motion is very restricted because it forces the audience to observe the scene from predefined viewing areas.

2.3.2. Reach-in systems

Reach-In Systems [61, 104, 91, 129] are desktop configurations that normally consist of an upside-down CRT screen which is reflected by a small horizontal mirror. They can be considered as screen-based augmented reality systems which provide optical see-through. Nowadays, these systems present stereoscopic 3D graphics to a single user who is able to reach into the presented visual space by directly interacting below the mirror while looking into the mirror. Thus, occlusion of the displayed graphics by the user's hands or input devices is avoided. Such systems are used to overlay the visual space over the interaction space, whereby the interaction space can contain haptic information rendered by a force-feedback device such as a PHANTOM [72]. While most reach-in systems apply full mirrors [91, 129], some utilize half-silvered mirrors to augment the input devices with graphics [61, 104] or temporarily exchange the full mirror by a half-silvered one for calibration purposes [129].

Knowlton [61], for instance, overlaid monoscopic 2D keycap graphics on the user's view of an otherwise conventional keyboard by using a half-silvered mirror that reflected a CRT screen. This allowed the graphics to annotate the user's fingers within the illuminated workspace below the mirror instead of being blocked by them.

Schmandt's Stereoscopic Computer Graphic Workstation [104] is another early example of a reach-in arrangement that applies an electro-magnetic tracking device for input in combination with a CRT screen and a half-silvered mirror. He superimposed 3D graphics over the transmitted image of the working area below the mirror.

Poston and Serra [91] developed the Virtual Workbench, but used a mechanical input device to overcome the magnetic field distortion problems of Schmandt's setup, which were caused by the interference between the CRT screen and the electro-magnetic tracking device.

A more recent development is the apparatus by Wiegand, Schloerb and Sachtler [129] (also named Virtual Workbench). Their system offers a trackball for input, a Phantom for input and additional force feedback, and stereo speakers for auditory feedback.

Due to the small working volume of these devices, their applications are limited to near-field operations. Although some of these systems employ half-silvered mirrors instead of full mirrors for calibration purposes, only a few support augmented reality tasks. The maturity of systems, however, renders exclusively virtual (visual and haptic) information. Several of these devices are commercially available (e.g., the Reach-In Display by Reach-In Technologies [102] or the Dextroscope by the Medical Imagine Group Med [91]) and are mainly used for medical/industrial simulation and training, or psychophysics and training research [129].

Similar to Pepper's ghost configurations, single planar mirrors with a static screen-mirror alignment (e.g., 30°-45°) are applied for reach-in systems - providing only one correct perspective (i.e., in case of reach-in systems for a single viewer, only). Although reach-in systems mostly present stereoscopic 3D graphics, a view-dependent rendering is normally not applied, since the user's head movements are naturally constrained by the near-field system itself.

2.3.3. Real image displays

Real Image Displays [31, 111, 79, 127, 114, 24, 73, 74, 28] are display systems that consist of single or multiple concave mirrors. Again, they belong to the class of re-imaging displays. Two types of images exist in nature -real and virtual. A real image is one in which light rays actually come from the image. In a virtual image, they appear to come from the reflected image - but do not. In case of planar or convex mirrors the virtual image of an object is behind the mirror surface, but light rays do not emanate from there. In contrast, concave mirrors can form reflections in front of the mirror surface where emerging light rays cross - so called "real images". Several real image displays are commercially available (e.g. [28]), and are mainly employed by the advertising or entertainment industry. On the one hand, they can present real objects that are placed inside the system so that the reflection of the object forms a three-dimensional real image floating in front of the mirror. On the other hand, a projection screen (such as a CRT or LCD screen, etc.) can be reflected instead -resulting in a free-floating two-dimensional image in front of the mirror optics that is displayed on the screen (some refer to these systems as "pseudo 3D displays" since the free-floating 2D image has an enhanced 3D quality). Usually, prerecorded video images are displayed with such real image displays.

One fundamental point of optical see-through AR is to use optical combiners to superimpose the real environment with reflected graphics. This requires that the displayed virtual objects appear within the same spatial space as the real objects to be augmented. However, if the real environment was located within the same spatial space as the real image formed by a real image display (i.e., in front of the mirror surface), these objects would occlude the mirror optics and consequently the reflected image. Thus, if virtual objects have to be superimposed over real ones, real image displays suffer from similar occlusion problems as regular projection screens. The second distinction of optical see-through AR to real image displays is that they usually do not make use of stereopsis and, in addition, are normally not able to dynamically display different view-dependent perspectives of the graphically presented scene. Note that some approaches apply additional optical elements (lenses) to cause an autostereoscopic viewing for a static viewpoint (i.e., a very limited viewing area) [73, 74].

2.3.4. Varifocal mirror displays

Varifocal Mirror Displays [122, 39, 73, 74] apply flexible mirrors and belong to the class of volumetric displays. In some systems the mirror optics is set in vibration by a rear-assembled loudspeaker [39]. Other approaches utilize a vacuum source to manually deform the mirror optics on demand to change it's focal length [73, 74]. Vibrating devices, for instance, are synchronized with the refresh-rate of a display system that is reflected by the mirror. Thus, the spatial appearance of a reflected pixel can be exactly controlled - yielding images of pixels that are displayed approximately at their correct depth (i.e., they provide an autostereoscopic viewing and consequently no stereo-separation is required). Due to the flexibility of varifocal mirror displays, their mirrors can dynamically deform to a concave, planar, or convex shape (generating real or virtual images). However, these systems are not suitable for optical see-through tasks, since the space behind the mirrors is occupied by the deformation hardware (i.e., loudspeakers or vacuum pumps). In addition, concavely shaped varifocal mirror displays face the same problems as real image displays. Therefore, only full mirrors are applied in combination with such systems.

2.4. Augmented reality displays

Displays are image-forming systems that apply a set of optical, electronic and mechanical components to generate images somewhere on the optical path in-between the observer's eyes and the physical object to be augmented. Depending on the optics being used, the image can be formed on a plane or on a more complex non-planar surface.

Figure 2.6 illustrates the different possibilities of where the image can be formed, where the displays are located with respect to the observer and the physical object, and what type of image is produced (i.e., planar or curved).

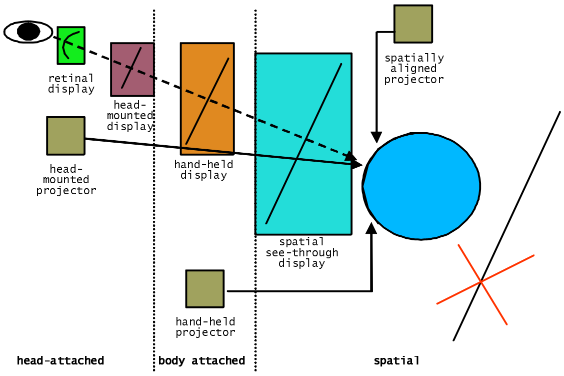


Figure 2.6: *Image-generation for augmented reality displays.*

Head-attached displays, such as retinal displays, head-mounted displays, and head-mounted projectors have to be worn by the observer. While some displays are hand-held, others are spatially aligned and completely detached from the users. Retinal displays and several projector-based approaches form curved images – either on the observer's retina or directly on the physical object. Most of the displays, however, form images on planes – called image-planes – that can be either head-attached or spatially aligned. Images behind real objects cannot be formed by a display that is located in front of real objects. In addition, if images are formed behind a real object, this object will occlude the image portion that is required to support augmentation.

Several pros and cons can be found by comparing the different types of displays. Most of them will be discussed within the following section.

If stereoscopic rendering is used to present mixed (real and virtual) worlds, two basic fusion technologies are currently being used: *video-mixing* and *optical combination*.

While video-mixing merges live record video streams with computer generated graphics and displays the result on the screen, optical combination generates an optical image of the real screen (displaying computer graphics) which appears within the real environment (or within the viewer's visual field while observing the real environment). Both technologies entail a number of advantages and disadvantages which influence the type of application they can address. A discussion on advantages and disadvantages of video-mixing and optical combination can be found in [103] and [3].

Today, most of the stereoscopic AR displays require to wear some sort of goggles to provide stereo separation. We believe, however, that auto-stereoscopic approaches will play a dominat role in the future of AR.

Several characteristics of augmented reality displays have been classified by Milgram et al [77, 78]:

- Provides an egocentric (immersive) or exocentric (non-immersive) experience;
- Maintains an orthoscopic (1:1) mapping between size and proportions of displayed images and real environment;
- Offers a direct or indirect view on the real environment.

In this section, we discuss several types of augmented reality displays. Note that we rather present the display categories that are relevant for alternative AR concepts, than to provide a complete list of individual devices.

2.4.1. Screen-based augmented reality

Screen-Based augmented reality has sometimes been referred to as *window on the world* [37]. Such systems make use of video-mixing and display the merged images on a regular monitor. According to Milgram's classification [77, 78], traditional screen-based augmented reality displays are exocentric, non-orthoscopic and provide a remote view on the real environment.

As fish tank VR systems which also apply monitors, window on the world setups provide a low degree of immersion. Within an augmented reality context the degree of immersion into an augmented real environment is frequently expressed by the amount of the observer's visual field (i.e., the field of view) that can be superimposed with graphics. In case of Screen-Based Augmented Reality, the field of view is limited and restricted to the monitor size, its spatial alignment relative to the observer, and its distance to the observer.

For Screen-Based Augmented Reality, the following disadvantages can be found:

- Small field of view that is due to relatively small monitor sizes;
- Limited resolution of the merged images (especially dissatisfying is the limited resolution of the real environment);
- Mostly provides a remote viewing, rather than supporting a see-through metaphor;
- Direct interaction with the real environment and the graphical augmentation is usually not supported.

The latter two points do not apply to reach-in configurations – no matter if they are realized with see-through LCD panels (e.g., [105]), mirror-beam splitters (e.g., the ones described in section 2.3.2, or the one shown in figure 2.7), or other optical combiners.

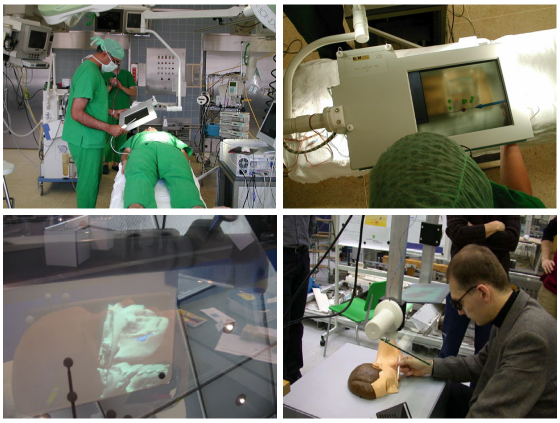


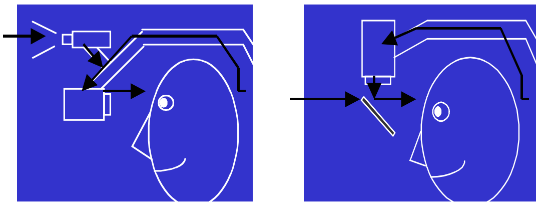
Figure 2.7: *Example of a screen-based AR display using a see-through LCD panel – The AR window (upper-left and -right) [105]. Courtesy: ZGDV, Schwald. Example of a mirror-based AR display using optical see-through beam splitters (lower-left and -right). Courtesy: Fraunhofer IMK (www.arsys-tricorder.de).*

Another example of a multi-user screen-based augmented reality display is the variation of the Virtual Showcase [13] that is illustrated in figure 2.5. It does not support direct interaction, but follows an optical see-through concept.

Note that screen-based augmented reality can be extended to large projection screens [94]. In this case, the shortcomings that are related to a small field of view, a limited resolution, and a remote viewing can be suspended (i.e., they can also be egocentric and orthoscopic).

2.4.2. Head-mounted displays

Head-mounted displays are currently the display devices which are mainly used for augmented reality applications. They belong to the category of head-attached displays.



video see-through optical see-through

Figure 2.8: *Video see-through and optical see-through [3]. Courtesy: Azuma.*

Two different head-mounted display-technologies exist to superimpose graphics onto the user's view of the real world: *Video see-through head-mounted displays* that make use of video-mixing and display the merged images within a closed-view head-mounted display, or *optical see-through head-mounted displays* that make use of optical combiners (essentially half-silvered mirrors or transparent LCD displays). With respect to Milgram's classification [77, 78], video and optical see-through head-mounted displays are egocentric, orthoscopic and provide a direct (optical see-through) or indirect (video see-through) view on the real environment.

However, several disadvantages can be related to the application of head-mounted displays as an augmented reality device. Note that most of these shortcomings are inherited form the general limitations of head-attached display technology:

- Lack in resolution that is due to limitations of the applied miniature displays;
- Limited field of view that is due to limitations of the applied optics;
- Imbalanced ratio between heavy optics (that results in cumbersome and uncomfortable devices) and ergonomic devices with a low image quality;
- Visual perception issues that are due to the constant image depth. Since objects within the real environment and the image plane that is attached to the viewer's head are sensed at different depths, the eyes are forced to either continuously shift focus between the different depth levels, or perceive one depth level unsharp. This is known as the fixed focal length problem, and is more critical for see-through than for closed-view head-mounted displays;
- Require difficult user and session independent calibration;
- Increased incidence of discomfort due to simulator sickness because of head-attached image plane (especially during fast head movements) [88].

Head-mounted displays are currently the dominant display technology within the AR field. They support mobile applications and multi-user applications, if a large number of users need to be supported.

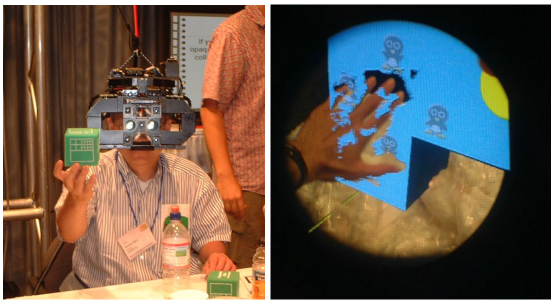


Figure 2.9: *Osaka University's ELMO – An optical see-through head-mounted display that provides mutual occlusion by using a see-through LCD panel in front of the HMD optics* [58]*. Courtesy: Kiyokawa.*

Some variations of head-mounted displays exist that are more attached to the real environment than to the user. Optical see-through boom-like displays (e.g., Osaka University's ELMO [58]) or video see-through, application-adopted devices (e.g., the head-mounted operating microscope [38]) represent only two examples.

2.4.3. Projection-based augmented reality (PBAR)

In general, a *projection-based augmented reality (PBAR)* [14] configuration is a spatial projection screen (or multiple screens) that is enhanced with optical see-through technology and supports stereoscopic, view-dependent and off-axis viewing of a graphically superimposed real environment.

In particular, we can characterize PBAR configurations to have the following properties:

- They combine optical see-through technology with spatial projection screens;
- Half-silvered mirror-beam splitters are applied as optical combiners (although the proposed concept can be extended toward other optical combination technologies, this work focuses on half-silvered mirror beam-splitters);
- They support the application of single or multi-faced planar optics as well as curved optics;
- They apply convexly curved and/or planar mirrors that form virtual images (although the proposed rendering techniques also support concave mirrors);
- They support static as well as flexible mirror-screen alignments;
- They provide a view-dependent image presentation, to dynamically display different perspectives of the presented scene;
- They represent general off-axis optical systems (however, the special on-axis case is included);
- They simultaneously support single or multiple observers;
- They apply several rendering and image transformation techniques that compensate for the optical effects that are produced by the elements of a PBAR configuration. These optical effects include reflection-deformations caused by mirrors, refraction-distortion caused by lenses (i.e., semi-transparent mirror-beam splitters), and optical distortion caused by miscalibrated displays;
- They require interactive stereoscopic rendering to make use of stereopsis.

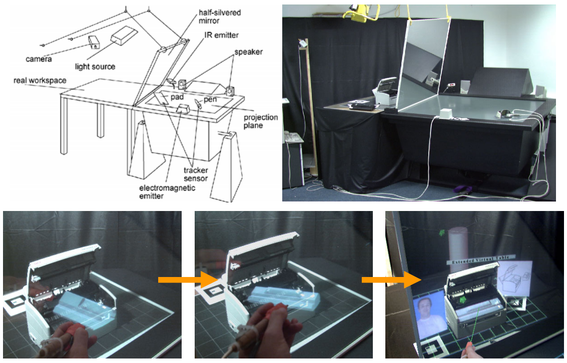


Figure 2.10: *Example of a PBAR configuration – The Extended Virtual Table* [11]*.*

The following general shortcomings are related to PBAR configurations:

- 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.

Examples for PBA configurations are the Extended Virtual Table [11] and variations of the Virtual Showcase [10], such as the one shown in figure 2.11.

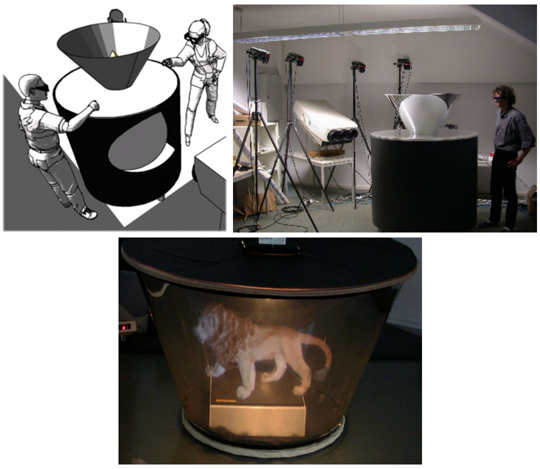


Figure 2.11: *Another example of a PBAR configuration – The Virtual Showcase (cone-shaped variation)* [10].

2.4.4. Spatially augmented reality (SAR) and everywhere displays

Spatially augmented reality [96, 97] is another alternative to head-mounted displays with the same core idea than PBAR -namely to embed the display technology into the real environment. We can say that it represents an extreme case of oblique screen displays.

In Spatially augmented reality, front-projection devices are used to seamlessly project images directly on physical objects' surfaces instead of displaying them somewhere within the viewer's visual field, as it is done with head-mounted displays. With respect to Milgram's classification [77, 78], spatially augmented reality is egocentric, orthoscopic and provides a direct view on the real environment. A stereoscopic projection and consequently the technology to separate stereo images is not necessarily required if only the surface properties (e.g., its color, illumination or texture) of the real objects are changed by overlaying images [97]. In this case a correct depth perception is still provided by the physical depth of the objects' surfaces. This is similar to the notion of volumetric displays that also directly illuminate spatial points within a display volume to provide an autostereoscopic viewing.

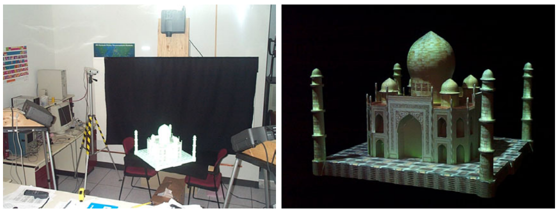


Figure 2.12: *Example of a Spatially AR approach – Shader Lamps* [99].

However, if 3D graphics are displayed in front of the object's surfaces, a view-dependent, stereoscopic projection is required as for other oblique screen displays.

On the one hand, this overcomes some of the shortcomings that are related to head-mounted displays: an improved ergonomics, a theoretically unlimited field of view, a scalable resolution, and an easier eye accommodation (because the virtual objects are typically rendered near their real world location).

On the other hand spatially augmented reality introduces several new problems:

- Shadow-casting of the physical objects and of interacting users that is due to the utilized front-projection;
- Restrictions of the display area that is constrained to the size, shape, and color of the physical objects' surfaces (for example, no graphics can be displayed beside the objects' surfaces);
- Restricted to a single user in case virtual objects are displayed with non-zero parallax.

spatially augmented reality detaches the display device from the user.

Similar to SAR, the Everywhere Displays project aims to develop systems that allow the transformation of every surface in a space into a projected "touch screen". A Prototype that combines a LCD projector, a pan/tilt mirror, and a camera has been presented in [90].

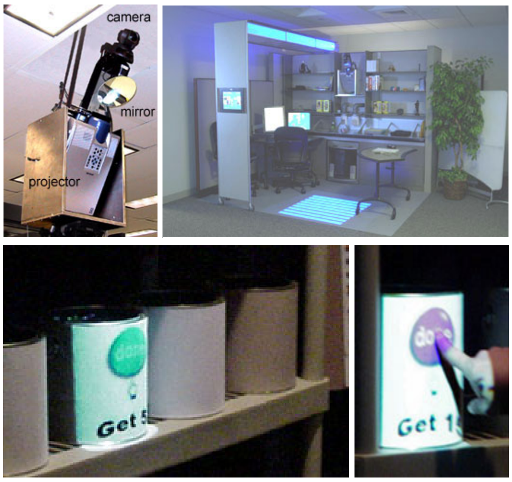


Figure 2.13: *The Everywhere Display concepts* [90]. *Courtesy: IBM, Pinhanez.*

The mirror is used to deflect the image of the projector to surfaces, walls, or the floor of a room. The projected image can be processed to compensate for the perspective distortion. A pan/tilt video camera is employed to detect hand/body activity on the projected area, so people can interact with the projected image by simply touching the surface. In contrast to SAR, the projection frustum can be dynamically changed with the everywhere display concept. However, both consents share the same limitations that have been listed above.

2.4.5. Transparent projection screens

In contrast to traditional front or rear-projection systems that apply opaque canvases or ground glass screens, *transparent projection screens* don't block the observer's view to the real environment behind the display surface. Therefore, they can be used as optical combiners that overlay the projected graphics over the simultaneously visible real environment. According to Milgram's classification [77, 78], transparent projection screens are semi-egocentric, orthoscopic and provide a direct view on the real environment. They belong to the category of embedded screen displays.

Pronova's HoloPro system [92] is such a transparent projection screen. It consists of a multi-layered glass plate that has been laminated with a light-directing holographic film. The holographic elements on this film route the impinging light rays into specific directions, rather than to diffuse them into all directions (as it is the case for traditional projection screens). This results in a viewing volume of 60° horizontal and 20° vertical range in front of the screen, where the projected images are visible. Regular projectors can be used to rear-project onto a HoloPro screen. However, they have to beam the images from a specific vertical angle (36.4°) to let them appear within the viewing volume. Originally, the HoloPro technique has been developed to support bright projections at daylight.

Several shortcomings (mainly due to the applied holographic film) can be related to this technology:

- Limited and restricted viewing area;
- Static and constrained alignment of projector and projection plane (and therefore no flexibility);
- Low resolution of the holographic film (the pattern of the holographic elements are well visible on the projection plane);
- Reduced see-through quality due to limited transparency of non-illuminated areas.

Some researchers already begin to adapt this technology for augmented reality purposes [84]. The application of transparent projection screens for augmented reality also offers the potential to spatially embed optical see-through display technology into the real environment. Although multi-plane configurations are imaginable (but not yet realized), curved transparent projection screens do not exist and will be difficult to produce with holographic films that route the impinging light rays into specific directions.

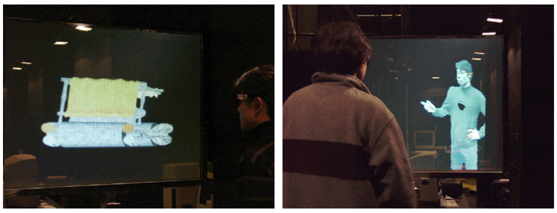
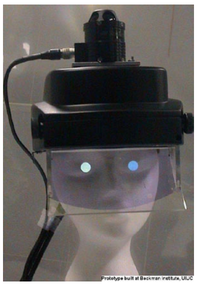


Figure 2.14: *Ogi's invisible interface* [84]. *Courtesy: Ogi, Yamada, Yamamoto, and Hirose.*

2.4.6. Head-mounted projectors

Head-mounted projective displays (HMPDs) [87, 50, 48] or *projective head-mounted displays (PHMDs)* [56] have recently been introduced as an alternative to head-mounted displays. Both devices apply head-mounted miniature projectors (LCD projectors or laser projectors), to beam the generated images from a dynamically moving center of projection. Thus, they approach to match the projector's center of projection and its projection frustum with the viewer's viewpoint and her viewing frustum. By doing this, the displayed images always appear optically undistorted - even when projected onto complex non-planar surfaces.



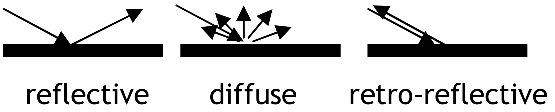
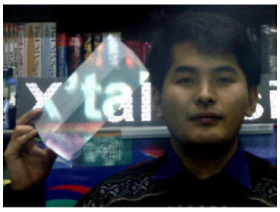


Figure 2.15: *Example of a head-mounted projector [48] (top). Courtesy: Hua, Gao, Brown, Ahuja, and Rolland. Reflection-properties of retro-reflective material (bottom).*

As head-mounted displays, we can count head-mounted projectors to the category of head-attached displays. With respect to Milgram's classification [77, 78], head-mounted projector displays are egocentric, orthoscopic and provide a direct view on the real environment.

Head-mounted projective displays [87, 50, 48] redirect the projection frustum with a mirror beam-splitter so that the images are beamed onto *retro-reflective* surfaces that are located in front of the viewer. A retro-reflective surface is covered with many thousands of micro corner cubes. Since each micro corner cube has the unique optical property to reflect light back along its incident direction, such surfaces reflect brighter images than normal surfaces that diffuse light. Note that this is similar in spirit to the holographic films used for transparent projection screens. However, these films are back-projected while retro-reflective surfaces are front-projected.

Projective head-mounted displays [56] beam the generated images onto regular ceilings, rather than onto special surfaces that face the viewer. Two half-silvered mirrors are used to integrate the projected image into the viewer's visual field so that the projectors' parameters match the viewer's parameters (i.e., projection/viewing center and frustum).



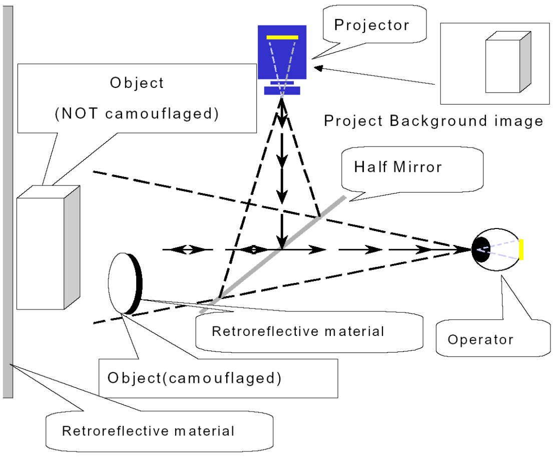


Figure 2.16: *Example of how HMPDs are used to make things transparent – Optical Camouflage [50]. Courtesy: Inami, Kawakami, Sekiguchi, Yanagida, Maeda, and Tachi.*

Similar to SAR, head-mounted projective displays decrease the effect of inconsistency of accommodation and convergence that is related to HMDs. Both, head-mounted projective displays and projective head-mounted displays also address other problems that are related to HMDs: They provide a larger field of view without the application of additional lenses that introduce distorting arbitrations. They also prevent incorrect parallax distortions caused by IPD (inter-pupil distance) mismatch that occurs if HMDs are worn incorrectly (e.g., if they slip slightly from their designed position). However, they also introduce several shortcomings:

- Both, head-mounted projective displays and projective head-mounted displays are heavy and highly cumbersome;
- Head-mounted projective displays inherit the shadow casting problem from front-projection systems;
- The integrated miniature projectors offer limited (and unscalable) resolution and brightness;
- Head-mounted projective displays might require special display surfaces (i.e., retro-reflective surfaces) to provide bright images;

- For projective head-mounted displays, the brightness of the images depends on the environmental light conditions;
- Projective head-mounted displays can only be used indoors, since they require the presence of a ceiling.

Although such displays technically tend to combine the advantages of projection displays with the advantages of traditional HMDs, their cumbersomeness currently prevents them from being applicable. As head-attached displays in general, they suffer from the imbalanced ratio between heavy optics (or projectors) that results in cumbersome and uncomfortable devices or ergonomic devices with a poor image quality.

2.4.7. Hand-held and object-oriented displays

A *hand-held display* for real-time tomographic reflection has been introduced by Stetton, et al. [112].

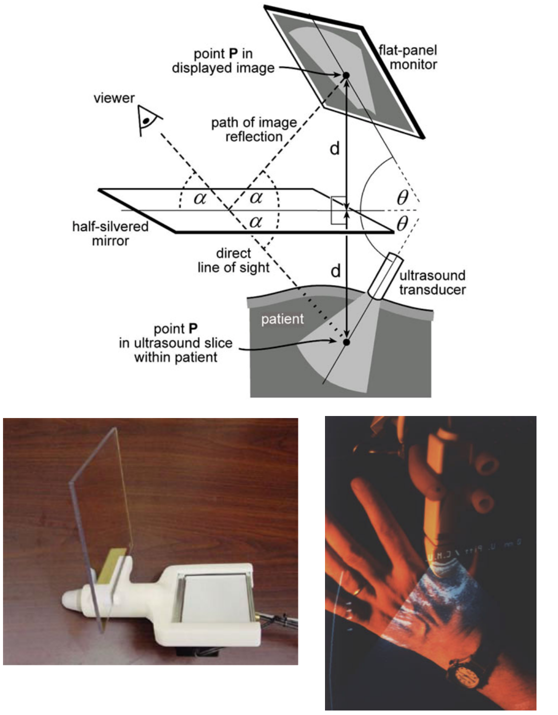


Figure 2.17: *Example of a hand-held mirror display - The Sonic Flashlight* [112]*. Courtesy: Stetten, Chib, Hildebrand, and Bursee.*

It consists of an ultrasound transducer that scans ultrasound slices of objects in front of it. The slices are displayed time-sequentially on a small flat-panel monitor and are then reflected by a planar half-silvered mirror in such a way that the virtual image is exactly aligned with the scanned slice area. Stereoscopic rendering is not required in this case, since the visualized data is two-dimensional and appears at its correct three-dimensional location.

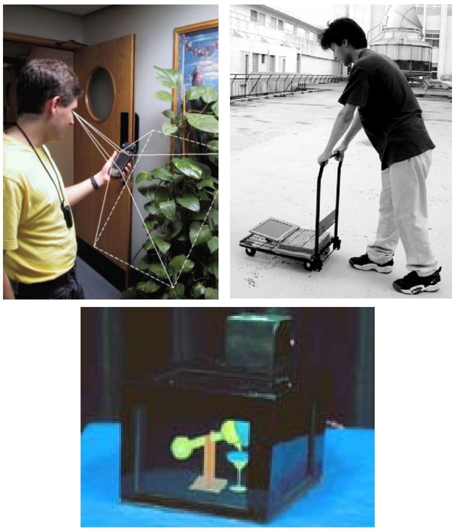


Figure 2.18: *Example of a hand-held PDA-based AR display* [82] *(upper-left). Courtesy: AT&T, Newman, Ingram, and Hopper. The Virtual Glassboat* [109] *(upper-right). Courtesy: Siio. Example of an object-oriented Display – MEDIA3* [54] *(bottom). Courtesy: Kawakami, Inami, Sekiguchi, Yanagida, Maeda, and Tachi.*

Other hand-held AR displays apply PDAs equipped with video cameras that allow supporting video see-through applications, or laptop-screens in a special configuration that allows looking underground [109].

Hand-held transflective surfaces (e.g., semi-transparent mirrors) can be used in combination with large, semi-immersive or immersive screens to support augmented reality tasks with rear-projection systems [8]. Tracked mirror beam-splitters act as optical combiners that merge the reflected graphics, which are displayed on the projection plane, with the transmitted image of the real environment.

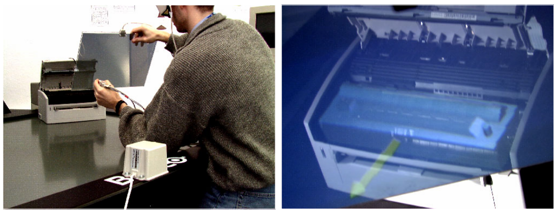


Figure 2.19: *The Transflective Pad [8] – a hand-held mirror beam-splitter in combination with a large rear-projection screen.*

While some optical see-through approaches apply optical elements, such as half-silvered mirror beam splitter or holographic projection screens, others use the real objects themselves as the displays (like SAR) or displays as the real objects (such as object-oriented displays).

MEDIA [54] is an example of an object-oriented display that consists of LCD panels, integrated into a box-like frame.

Yet another interesting display concept was described in [80] and proposes the application of a hand-held video projector as a real flashlight to interactively generate shadow effects of virtual objects on real surfaces. A combination of a hand-held video projector and a camera (cf. figure 2.19) is used by Foxlin and Naimark to demonstrate the capabilities of their optical tracking system. This concept might represent an interesting application of AR to the fields of architecture and maintenance.

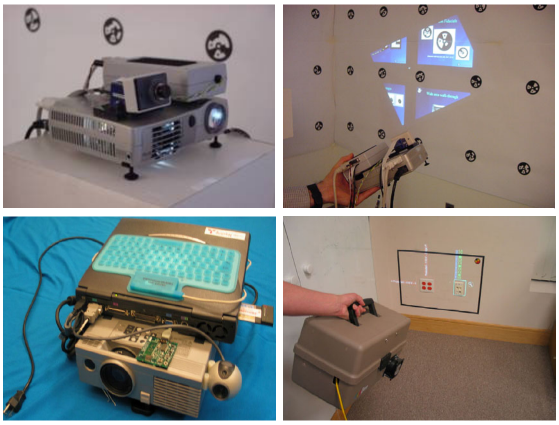


Figure 2.20: *AR Flashlight (top): augmenting the world with a tracked handheld projector. Courtesy: InterSense Inc., Foxlin and Naimark. Context aware iLamp (bottom): augmenting of an identified surface [101].*

A combination of a hand-held video projector and a camera (cf. figure 2.20-top) is used by Foxlin and Naimark to demonstrate the capabilities of their optical tracking system. It is also described by Raskar et al. in [101] (cf. figure 2.20 bottom).

3. Non-trivial projection screens

Chapter 3 describes augmented reality techniques using non-trivial projections screens. Our emphasis is on projector-based augmented reality. In this context, we describe two scenarios, a concept of spatial augmentation and then a more specific method of changing surface appearance of real objects.

3.1. Spatially augmented reality

In Spatially Augmented Reality (SAR), the user's physical environment is augmented with images that are integrated directly in the user's environment, not simply in their visual field [96, 97]. For example, the images could be projected onto real objects using digital light projectors, or embedded directly in the environment with flat panel displays. For the purpose of this discussion we will focus on projector-based augmented reality. While the approach has certain restrictions, it offers an interesting new method to realizing compelling illusions of virtual objects coexisting with the real world. The images could appear in 2D, aligned on a flat display surface, or they could be 3D and floating above a planar or even a non-planar surface. In the most basic applications, SAR combines the benefits of traditional spatially immersive displays and augmented reality displays.

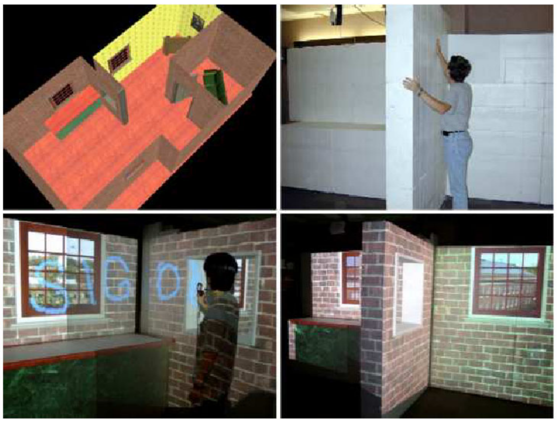


Figure 3.1: *Spatially augmenting large environment* [68]*. Virtual model (upper left). Physical display environment constructed using Styrofoam blocks (upper right). Augmented display (bottom). Note the view dependent nature of the display, the perspectively correct view through the hole in the wall and the windows. Courtesy: Low.*

The idea of SAR is best used when virtual objects are close to the physical objects on which they are displayed. For example, an architect can augment a tabletop scaled model of a house or building using a projector. She can start with a very simple neutral colored cardboard model and its geometric CAD representation. Then it is easy to add virtual objects such as door, windows, chimneys. She can also visualize underground water pipes or support structure inside the building. A compelling example of spatial augmentation is the application aimed at walk-thru of virtual human-sized environments built by Kok-lim Low et. al. in the Being There project at UNC [68]. Instead of building an exact detailed physical replica for projection, the display is made of simplified versions. For example, primary structures of building interiors and mid-sized architectural objects (walls, columns, cupboards, tables, etc.), can usually be approximated with simple components (boxes, cylinders, etc.). As seen in the Figure 3.1, display is made of construction Styrofoam blocks. The main architectural features that match the simplified physical model retain 3D auto-stereo, but the other details must be presented by projecting view-dependent images. Nevertheless, the experiment to simulate a building interior is convincing and provides a stronger sense of immersion when compared to SID, as the user is allowed to really walk around in the virtual environment. However, strategic placement of projectors to allow complete illumination and avoiding user shadows is critical.

3.2. Shader Lamps

In this section, we describe a special case of SAR. The idea is to replace a physical object with its inherent color, texture, and material properties with a neutral object and projected imagery, reproducing the original appearance directly on the object. Furthermore the projected imagery can be used to reproduce alternative appearances, including alternate shading, lighting, and even animation. The approach is to effectively lift the visual properties of the object into the projector and then re-project onto a neutral surface.

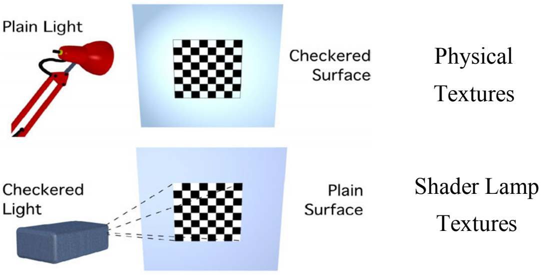


Figure 3.2: *The idea of ShaderLamps* [99]*.*

We use the phrase Shader Lamps to describe this mode of operation for projectors [99]. Consider the effect shown in Figure 3.3. The underlying physical object is a white diffuse vase. (The other objects such as the book and flowers are also real objects.)

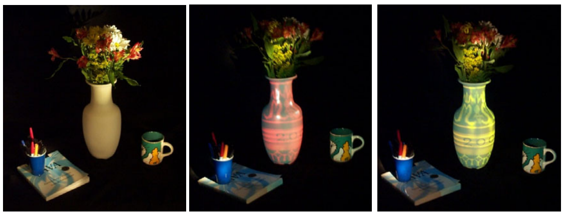


Figure 3.3: *The underlying physical object is a white diffuse vase (left). View-dependent effects, such as specular highlights, can be generated by tracking the user's location and projecting images on the vase (middle and right).* [99]

Can we make this white vase appear to look like it is made of marble, plastic or metal? Can we change the color or texture? The pictures show that the vase can be effectively 'painted' by projecting an image with view-independent diffuse shading, textures and intensity correction. The view-dependent effects such as specular highlights are generated for a given user location by modifying reflectance properties of the graphics model.

The figure shows appearance of a red plastic and a green metallic material on the clay vase. Although, there have been other attempts at augmenting appearances of objects by projecting color or texture, those effects are very limited and have been achieved for only specific applications. In this section, we show that the real challenges to realizing this as a new medium for computer graphics lies in addressing the problems related to complete illumination of non-trivial physical objects. The approach presented here offers a compelling method of visualization for a variety of applications including dynamic mechanical and architectural models, animated or living dioramas, artistic effects, entertainment, and even general visualization for problems that have meaningful physical shape representations. We present and demonstrate methods for using multiple Shader Lamps to animate physical objects of varying complexity, from a flower vase (Figure 3.3), to some wooden blocks, to a model of the Taj Mahal (Figure 3.4).

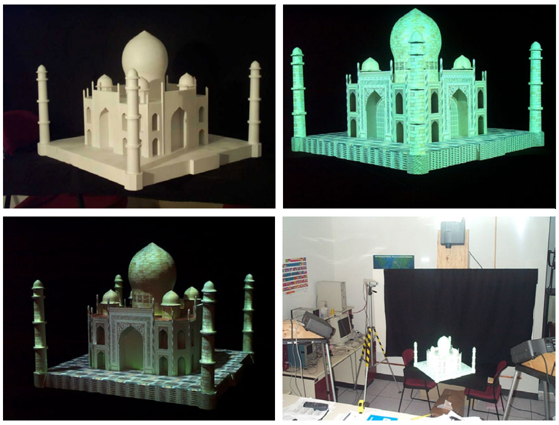


Figure 3.4: *ShaderLamps with TajMahal: The wooden white model (upper left) is illuminated. The scanned geometry of the TajMahal is authored to add texture and material properties. The geometry is registered to the real TajMahal is displayed from projector's viewpoint.* [99]

3.2.1. Illumination process

We introduce the idea of rearranging the terms in the relationship between illumination and reflectance to reproduce equivalent radiance at a surface. As shown in flatland in Figure 3.5, the radiance in a certain direction at point (x), which has a given BRDF in the physical world (left), can be mimicked by changing the BRDF and illuminating the point with a appropriately chosen light source, e.g. a projector pixel (right). Below we identify a radiance adjustment equation for determining the necessary intensity of a projector pixel, given the position and orientation of the viewer and the virtual scene. For a more systematic rendering scheme, we describe the notion of separating the *rendering view*—the traditional virtual camera view, from the *shading view*—the position of the viewer for lighting calculations.

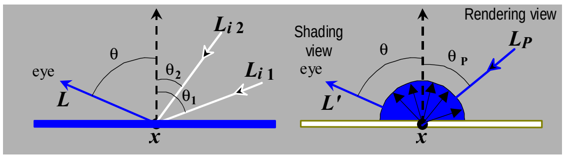


Figure 3.5: *The radiance at a point in the direction* (θ, ϕ) *(left). The radiance as a result of illumination from a projector lamp (right). By rearranging the parameters in the optical path, the two can be made equal.* [99]

First, let us consider the rendering equation, which is essentially a geometrical optics approximation as explained in [53]. The radiance at a visible surface point (x) in the direction (θ, ϕ) that would reach the observer of a physical realization of the scene is

$$L(x,\theta,\phi) = g(x,\theta,\phi)(L_e(x,\theta,\phi) + h(x,\theta,\phi)) \quad (3.1)$$

where

$$h(x,\theta,\phi) = \int_i F_r(x,\theta,\phi,\theta_i,\phi_i) L_i(x,\theta_i,\phi_i)\cos(\theta_i)d\omega_i \quad (3.2)$$

and $g(x,\theta,\phi)$ is the geometry term (visibility and distance), $L_e(x,\theta,\phi)$ is the emitted radiance at the point (non-zero only for light sources), and $F_r(x,\theta,\phi,\theta_i,\phi_i)$ is the BRDF of the point. The integral in $h(x,\theta,\phi)$ accounts for all reflection of incident radiance $L_i(x,\theta_i,\phi_i)$ from solid angles $d\omega_i$. Radiance has dimensions of energy per unit time, area and solid angle.

Treating the projector lamp as a point emitter, the radiance due to direct projector illumination at the same surface point at distance $d(x)$ but with diffuse reflectance $k_u(x)$ is given by

$$L'(x,\theta,\phi) = g(x,\theta,\phi)k_u(x)I_p(x,\theta_p,\phi_p)\cos(\theta_p)/d(x)^2 \quad (3.3)$$

where $I_p(x,\theta_p,\phi_p)$ = radiant intensity of projector in the direction (θ_p,ϕ_p) and is related to a discretized pixel value via filtering and tone representation.

We can reproduce radiance $L'(x,\theta,\phi)$ equivalent to $L(x,\theta,\phi)$ for a given viewer location, by solving Equation (3) for I_p:

$$I_p(x,\theta_p,\phi_p) = \frac{L(x,\theta,\phi)d(x)^2}{k_u(x)\cos(\theta_p)} \quad \text{for } k_u(x) > 0 \tag{3.4}$$

Thus, as long as the diffuse reflectance $k_u(x)$ is nonzero for all the wavelengths represented in $L(x,\theta,\phi)$, we can effectively represent the surface attribute with appropriate pixel intensities. In practice, however, the range of values we can display are limited by the brightness, dynamic range and pixel resolution of the projector.

The rendering process here involves two viewpoints: the user's and the projector's. A simple approach would be to first render the image as seen by the user, which is represented by $L(x,\theta,\phi)$, and then use traditional image-based rendering techniques to warp this image to generate the intensity-corrected projected image, represented by $I_p(x,\theta_p,\phi_p)$ [23, 75]. For a changing viewer location, view-dependent shading under static lighting conditions can also be implemented [27, 66, 42]. However, the warping can be avoided in the case where the display medium is the same as the virtual object. For a single-pass rendering, we treat the moving user's viewpoint as the shading view. Then, the image synthesis process involves rendering the scene from the projector's view, by using a perspective projection matrix that matches the projector's intrinsic and extrinsic parameters, followed by radiance adjustment. The separation of the two views offers an interesting topic of study. For example, for a static projector, the visibility and view-independent shading calculations can be performed just once even when the user's viewpoint is changing.

To realize a real-time interactive implementation we use conventional 3D rendering APIs, which only approximate the general rendering equation. The BRDF computation is divided into view-dependent specular, and view-independent diffuse and ambient components. View-independent shading calculations can be performed by assuming the rendering and shading view are the same. (The virtual shadows, also view-independent, are computed using the traditional two-pass shadow-buffer technique.) For view-dependent shading, such as specular highlights (Figure 3.5), however, there is no existing support to separate the two views. A note in the appendix describes the required modification.

3.2.2. Secondary scattering

Shader Lamps are limited in the type of surface attributes that can be reproduced. In addition, since we are using neutral surfaces with (presumed) diffuse characteristics, secondary scattering is unavoidable and can potentially affect the quality of the results. When the underlying virtual object is purely diffuse, sometimes the secondary scattering can be used to our advantage. The geometric relationships, also known as form factors, among parts of the physical objects, are naturally the same as those among parts of the virtual object. Consider the radiosity solution for a patch *i* in a virtual scene with *m* light sources and *n* patches:

$$\begin{aligned} B_{i\text{-intended}} &= k_{d_i}\sum_j B_j F_{i,j} = \\ &k_{d_i}\left(\sum_m B_m F_{i,m} + \sum_n B_n F_{i,n}\right) \end{aligned} \tag{3.5}$$

Here k_d is the diffuse reflectance, B_j is the radiance of patch *j*, and $F_{i,j}$ is the form factor between patches. Using Shader Lamps to reproduce simply the effect of direct illumination (after radiance adjustment), we are able to generate the effect of *m* light sources:

$$B_{i\text{-direct}} = k_{d_i}\sum_m B_m F_{i,m}. \tag{3.6}$$

However, due to secondary scattering, if the neutral surfaces have diffuse reflectance k_u, the perceived radiance also includes the secondary scattering due to the *n* patches, and that gives us

$$\begin{aligned} B_{i\text{-actual}} &= B_{i\text{-direct}} + B_{i\text{-secondary}} = \\ &k_{d_i}\sum_m B_m F_{i,m} + k_u \sum_n B_n F_{i,n}. \end{aligned} \tag{3.7}$$

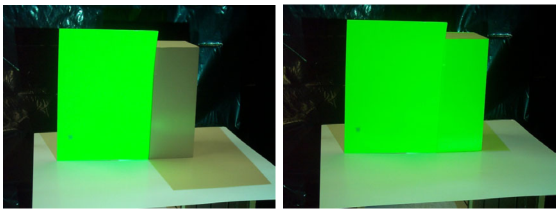


Figure 3.6: *A green paper illuminated with white light (left). The white diffuse surface on the right is illuminated with green light (right).* [99]

The difference between the desired and perceived radiance is

$$\left|(k_{d_i} - k_u)\sum_n B_n F_{i,n}\right|. \tag{3.8}$$

Thus, in scenarios where k_d and k_u are similar, we get approximate radiosity for "free"—projection of even a simple direct illumination rendering produces believable "spilling" of colors on neighboring parts of the physical objects. From the equation above, the secondary contribution from the neutral surfaces is certainly not accurate, even if we reproduce the first bounce exactly,. The difference is even larger when the virtual object has non-lambertian reflectance properties. We are currently investigating *inverse global illumination* methods so that the projected image can more accurately deliver the desired global illumination effect. Figure 3.6 shows a green and a white paper with spill over from natural white and projected green illumination. In this special case, the secondary scattering off the horizontal white surface below is similar for both parts.

3.3. Rendering techniques for non-trivial projection screens

3.3.1. Changing surface appearance of real objects

The image-based illumination of physical objects has been explored by many. But, we believe, two main challenges have kept the previous efforts to only expensive, large scale, or one-off implementations. (a) First, the geometric registration problem, which is cast as matching the projection of a single 2D image with an object. The projection of a perspective device has up to 11 degrees of freedom (6 external and 5 internal) [34], therefore, any effort to manually achieve the registration is likely to be extremely tedious. We propose a new simple technique below. (b) The second problem, which appears to be unexplored, is the complete illumination of non-trivial physical objects in presence of shadows due to self occlusion. With the advent of digitally-fed projectors and real-time 3D graphics rendering, a new approach for image-based illumination is now possible. We approach these problems by creating a 3D geometric understanding of the display setup. We describe an important intensity correction step and our solution for dealing with shadows.

3.3.2. Authoring and alignment

One of the important tasks in achieving compelling visualization is to create the association between the physical objects and the graphics primitives that will enhance those objects when projected. For example, how do we specify which texture image should be used for the face of a building model, or what color distribution will look better for a physical object? We need the physical object as well as its geometric 3D representation, and real or desired surface attributes. As mentioned earlier, many hardware and software solutions are now available to scan/print 3D objects and capture/create highly detailed, textured graphics models. The authoring can also be done interactively by "painting" directly on top of the physical objects. The result of the user interaction can be projected on the objects and also stored on the computer. Ideally, a more sophisticated user interface would be used to create and edit graphics primitives of different shape, color and texture.

To align a projector, first we *approximately* position the projector and then *adapt* to its geometric relationship with respect to the physical object. That relationship is computed by finding projector's intrinsic parameters and the rigid transformation between the two coordinate systems. This is a classical computer vision problem [34]. As seen in Figure 3.7, we take a set of fiducials with known 3D locations on the physical object and find the corresponding projector pixels that illuminate them. This allows us to compute a 3×4 perspective projection matrix up to scale, which is decomposed to find the intrinsic and the extrinsic parameters of the projector. The rendering process uses the same internal and external parameters, so that the projected images are registered with the physical objects.

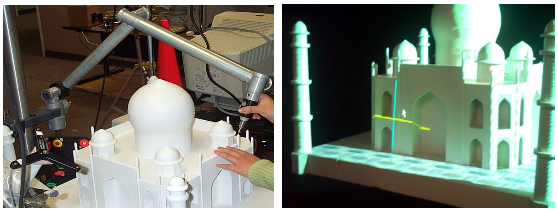


Figure 3.7: *We use a 3D touch probe scanner to create a 3D model of the real object (left). The projectors are calibrated with respect to the model by finding which pixels (center of cross) illuminate the known 3D points (right).* [99]

3.3.3. Intensity correction

The intensity of the rendered image is modified on a per-pixel basis to take into account the reflectance of the neutral surface, the local orientation and distance with respect to the projector using Equation (4). Since the surface normals used to compute the *$1/cos(\theta_P)$* correction are available only at the vertices in polygonal graphics models, we exploit the rendering pipeline for approximate interpolation. We illuminate a white diffuse version of the graphics model (or a model matching appropriate *$k_u(x)$* of the physical model) with a virtual white light placed at the location of the projector lamp and render it with squared distance attenuation. The resultant intensities are smooth across curved surfaces due to shading interpolation and inversely proportional to *$(d(x)^2/k_u(x)cos(\theta_P))$* factor. To use the limited dynamic range of the projectors more efficiently, we do not illuminate surfaces with $\theta_P>60$ (since *$1/cos(\theta)$* ranges from 2 to infinity). This avoids the low sampling rate of the projected pixels on oblique surfaces and also minimizes the misregistration artifacts due to any errors in geometric calibration. During the calculations to find the overlap regions (described below), highly oblique surfaces are considered not to be illuminated by that projector.

3.3.4. Summary of operations

```
during pre-processing:
        create 3d graphics model, G, of physical object
        create 3d graphics model, B, of background
        approximately position the projector
        find perspective pose, P, of the projector wrt the
        physical object

   during run-time:
        get user location, U
        get animation transformation, T
        modify G's surface attributes
        render G using the pose P, and user location U
        transform B using T-1, B'
        render B' using the pose P, and user location U
        modify image intensity to compensate for surface
        orientation
```

Let is look at the steps in detail. The user's location, *U*, can be tracked using magnetic or optical tracking technology. To reproduce purely view-independent surface appearance (diffuse reflectance), user location is not required. For view-dependent effects such as specular highlights, approximate user location is necessary. We may assume the user is at a sweet-spot and need not track the user. The projector projection matrix, *P*, is obtained using an off-line calibration process similar to the technique used for finding internal and external parameters of a camera [34]. We take a set of fiducials with known 3D locations on the physical object and find the corresponding projector pixels that illuminate them. This allows us to compute a 3x4 perspective projection matrix up to scale, which is decomposed to find the internal and the external parameters of the projector. The rendering process uses the same internal and external parameters, so that the projected images are registered with the physical objects. During run-time, instead of the object, *G,* the background, *B*, is transformed to create the apparent motion. At each frame, an intensity correction stage pre-multiplies the projected image with intensity weights that compensate for the local surface orientation. Otherwise, surfaces normal to the incident light will appear brighter than surfaces illuminated obliquely due to the cosine fall-off.

3.4. Complete illumination of complex 3D shapes

For complete illumination that avoids shadows due to self-occlusion, using additional projectors is an obvious choice. This leads to the more difficult problem of seamlessly merging images from multiple projectors. A naïve solution may involve letting only a single projector illuminate any given surface patch. But, there are two main issues when dealing with overlapping CRT, LCD or DLP projectors, which compel the use of feathering (or cross-fading) of intensities. The first is the lack of color equivalence between neighboring projectors [71], due to manufacturing process and temperature color drift during their use. The second is our desire to minimize the sensitivity to small errors in the estimated geometric calibration parameters or mechanical variations.

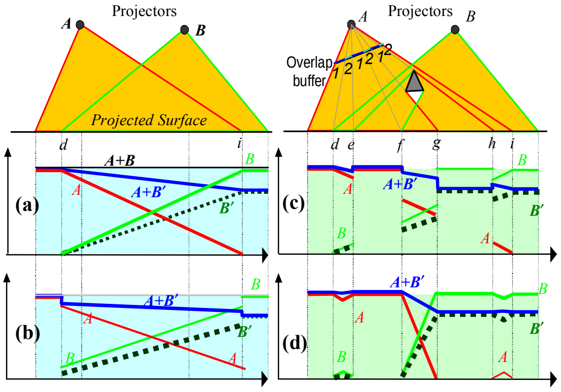


Figure 3.8: *Intensity weights using feathering methods. The plots show the contribution of projectors A, B and B′ and the resultant accumulation A+B and A+B′ along the lit planar surface. Our technique, shown in (d), creates smooth weight transitions.* [99]

Feathering is commonly used to generate seamless panoramic photomosaics by combining several views from a single location [118]. Similar techniques are exploited in multi-projector wide-field-of-view displays [86, 97], and two-dimensional arrays of flat projections. In such cases, the overlap region is typically a (well-defined) contiguous region on the display surface as well as in each projector's frame buffer. In the algorithm used in [118, 97] the intensity of a pixel is weighted proportional to the Euclidean distance to the nearest boundary (zero contribution) pixel of the (projected) image. The per-pixel weights are in the range [0, 1]. They are multiplied to the pixel intensities in the final rendered image. The pixels weights near the boundary of a source image are near zero and the pixels contribute very little, so that there is a smooth transition to the next source image. This leads to the commonly seen intensity roll-off as shown in Figure 3.8(a). Under ideal conditions and assuming color equivalence, the weight contribution of both projectors *A*+*B* adds up to 1. Even when projector *B*'s color response is different than that of *A* (say, attenuated—shown as *B*′), the resultant *A*+*B*′ (shown in blue) transitions smoothly in the overlap region.

This weight assignment strategy works well only when the target image illuminates a smooth continuous surface at and around the overlap. In our case, the physical model is usually made up of non-convex objects or a collection of disjoint objects resulting in shadows, fragmented overlap regions and, more importantly, overlap regions containing surfaces with depth discontinuities, as shown in Figure 3.8(c) with a simple occluder. Now, with unequal color response, the resultant weight distribution A+B′ has offending sharp changes, e.g. at points f and g. This situation is analogous to image-based rendering (IBR), where warping a single depth-enhanced image creates dis-occlusion artifacts. When multiple source images are warped to the target image, the color assigned to a pixel needs to be derived (from either a single image where they overwrite each other or) as a weighted combination of corresponding pixels from source images. The feathering, which actually blurs the result, is usually necessary to overcome (minor) color difference in corresponding pixels in input images and to hide ghosting effects (due to small mis-registration errors). One of the few solutions to this is proposed by [27], in which they scale the intensities by weights proportional to the angles between the target view and the source views. As mentioned in their paper, "it does not guarantee that the weights will transition smoothly across surfaces of the scene. As a result, seams can appear in the renderings where neighboring polygons are rendered with very different combinations of images." The plots in Figure 3.8(b) show a sample weighting scheme based on a similar idea and the corresponding problems. Below, we present a global solution using a new feathering algorithm that suits IBR as well as Shader Lamps. The algorithm is based on the following guidelines:

1. The sum of the intensity weights of the corresponding projector pixels is one so that the intensities are normalized;
2. The weights for pixels of a projector along a physical surface change smoothly in and near overlaps so that the inter-projector color differences do not create visible discontinuity in displayed images; and
3. The distribution of intensity weights for a projector within its framebuffer is smooth so that small errors in calibration or mechanical variations do not result in sharp edges.

In practice, it is easier to achieve (or maintain) precise geometric calibration than to ensure color equality among a set of projectors over a period of time [71]. This makes condition (2) more important than (3). But, it is not always possible to satisfy condition (2) or (3) (e.g. if the occluder moves closer to the plane so that f = g in Figure 3.8) and hence they remain as guidelines rather than rules.

The three guidelines suggest solving the feathering problem, without violating the weight constraints at depth discontinuities and shadow boundaries. Traditional feathering methods use the distance to the nearest boundary pixel to find the weight [118, 97]. Instead, we first find pixels corresponding to regions illuminated by a single projector and assign them an intensity weight of 1. Then, for each remaining pixel, the basic idea behind our technique is to find the shortest Euclidean distance to a pixel with weight 1, ignoring paths that cross depth discontinuities. The assigned weight is inversely proportional to this distance. Figure 6(d) shows the result of the new feathering algorithm in flatland for two projectors. Even under different color responses, the algorithm generates smooth transitions (see A+B′) on the planar surface in presence of shadows and fragmented overlaps. The algorithm can be used for 3 or more projectors without modification.

For a practical implementation, we use two buffers—an overlap buffer and a depth buffer. The depth buffer is updated by rendering the graphics model. The overlap buffer contains integer values to indicate the number of overlapping projectors for each pixel. The overlap regions (i.e. overlap count of two or more) are computed using the traditional shadow-buffer technique. The algorithm follows:

```
at each projector:
    compute boundaries between regions of overlap
    count 1 and > 1
    compute depth discontinuities using edge
    detection in depth buffer
        for each pixel in overlap region
            update shortest distance to
            overlap count ==1 region
            ignoring
            paths crossing depth
            discontinuity
        endfor
at each projector:
    for each pixel in overlap region
        find all corresponding pixels in other
        projectors
        assign weights inversely proportional
        to the shortest distance
    endfor
```

For some pixels in the overlap region, such as region [*h,i*] for projector *A*, no nearest pixel with overlap count of 1 can be found, and so the shortest distance is set to a large value. This elegantly reduces the weight in isolated regions and also cuts down unnecessary transition zones.

3.5. Enhancing moving objects

We can illuminate objects so that the surface textures appear glued to the objects even as they move. In this case, we can display updated specular highlights even for a static viewer. For example, in showroom windows or on exhibition floors, one can show a rotating model of the product in changing colors or with different features enhanced. In an experimental system, a tracked "paintbrush" was used to paint on a tracked moving cuboid held by the user (Figure 3.9). The presence of the physical model allows natural haptic feedback. The need to attach a tracker and dynamic mis-registration due to tracker latency are the two main problems [19].

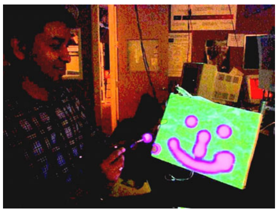


Figure 3.9: *A tracked "paintbrush" painting on a tracked cuboid* [19].

4. Spatial optical see-through displays

Two different examples for spatial optical see-through displays are presented in section 4.1: The Extended Virtual Table as a single user setup, and the Virtual Showcase that supports multiple users. These examples will serve as references for describing the following rendering and illumination techniques.

Interactive rendering techniques that can be used to drive spatial optical see-through AR displays which consist of a combination of single or multiple beam-splitters (planar or curved) and screens (projection displays or monitors) are discussed in section 4.2.

The projector-based illumination concept is described in section 4.3, and examples are outlined how it can be used to create consistent illumination and occlusion effects.

Note that some of the described rendering and illumination methods utilize multi-pass techniques. Today's graphics hardware offers render-to-texture operations and procedural shaders that can significantly speed up the rendering performance. The presented algorithms describe our techniques in a general way – without a focus on implementation specific details.

4.1. Concepts and hardware prototypes

4.1.1. 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* [11]. A large half-silvered mirror is used as the optical combiner to unify a virtual and a real workbench. 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, and a seamless integration of such technology into habitual work environments.

4.1.1.1. Physical arrangement

Our Extended Virtual Table prototype consists of a virtual and a real workbench (cf. figures 2.10).

A Barco BARON [5] 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 are used to separate the stereo-images for both eyes and make stereoscopic viewing possible. In addition, an electro-magnetic tracking device 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.

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 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[*]. Higher-quality half-silvered mirrors with better optical characteristics could be used instead.

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 is applied to continuously capture a video-stream of the real workspace, supporting an optical tracking of paper-markers above the real workbench.

4.1.1.2. 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[†].

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.,

[*] Values valid if used with 6 mm thick regular glass.

[†] In our case the origin of the coordinate system is located at the center of the projection plane.

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.

How to render graphics if a single planar mirror optics is applied as optical combiner is described in section 4.2.1.3.

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). 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.

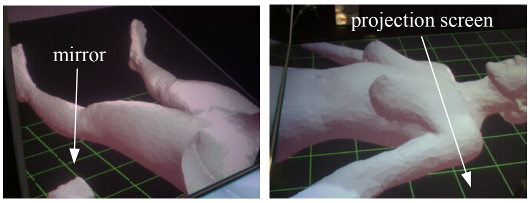


Figure 4.1: *A large coherent virtual content (a life-size human body for medical training) viewed in the mirror (left), or on the projection plane (right). The real workspace behind the mirror is not illuminated.* [11]

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 4.1 shows a large coherent virtual scene whose parts can be separately observed by either looking at the mirror or at the projection plane.

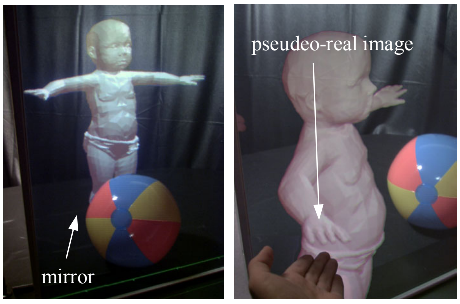


Figure 4.2: *Real objects behind the mirror -the ball- are illuminated and augmented with virtual objects -the baby- (left). The angle between mirror and projection plane is 60°. Without attaching a clipping plane to the mirror, the baby can reach its arm through the mirror (right). The angle between mirror and projection plane is 80°.* [11]

Figure 4.2 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[*].

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. 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 4.2-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.

[*] This is similar to the electronic shuttering of HMDs.

4.1.2. The Virtual Showcase

The Virtual Showcase [10] 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 Extended Virtual Table, Virtual Showcases offer the possibility to simultaneously support multiple head-tracked viewers and to provide a seamless surround view on the augmented real environment located within the showcase.

4.1.2.1. Physical arrangement

A Virtual Showcase consists of two main parts: a convex assembly of half-silvered mirrors and one or multiple graphics display.

Many different versions of Virtual Showcases have been developed so far. Some of them apply a large projection display and a curved, cone-shaped mirror optics (cf. figure 2.11), and others use multiple monitors in addition with planar pyramid-shaped mirror optics (cf. figure 2.5 and appendix 1).

While the pyramid-shaped prototypes simultaneously supports up to four viewers, the cone-shaped prototypes provides a seamless surround view onto the displayed artifact for less viewers.

Optional components of the Virtual Showcase are additional video projectors that are used for a pixel-precise illumination of the real content (e.g., to create consistent occlusion and illumination effects) and video cameras to receive feedback from the inside of the display. Techniques that apply these components are described in section 4.3.

4.1.2.2. General functioning

By assigning each viewer to an individual mirror, the pyramid-like Virtual Showcase prototype can support up to four observers simultaneously. Looking through the mirror optics allows to see the reflection of the corresponding screen portion at the same time, and within the same spatial space as the real object inside the showcase.

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 4.3-bottom

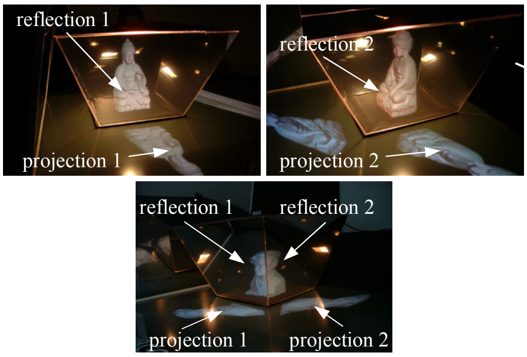


Figure 4.3: *Two individual views onto the same image space (top). Different reflections are optically merged into a single consistent image space (bottom).* [10]

Figures 4.3-upper left/upper 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.

How to render graphics if multiple planar mirrors are applied as optical combiners is described in sections 4.2.1.4 – 4.2.1.6.

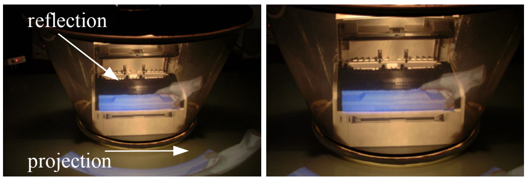


Figure 4.4: *The curved projection is reflected into a straight reflection.* [10]

Building Virtual Showcases from one single mirror 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 mirror optics requires a curvilinear of the image before it is displayed. How do warp the image for curved mirror beam-splitters is outlined in section 4.2.2.

4.2. Rendering techniques for spatial optical see-through displays

In this section we describe interactive rendering techniques for spatial optical see-through displays that can be applied for planar and curved mirror beam splitter. Depending on the applied optical combiner, these techniques have to neutralize physical reflection and refraction transformations of the projected graphics so that the optically formed images appear orthoscopic, stereoscopically and perspectively correct and undistorted to one or many observers.

The simplest spatial optical see-through configurations are transparent projection screens, such as the one presented in [84] or the see-through LCD panel described in section 2.4.1. A simple off-axis projection is sufficient to render graphical overlays perceptively correct for a head-tracked user. Consequently, the rendering techniques for such displays are exactly the same as for spatial VR displays (e.g., projection displays or simple monitors). They are not explicitly discussed in this tutorial. Instead, we will describe rendering extensions that support displays which apply additional mirror beam-splitters to achieve an optical combination.

4.2.1. Pipeline rendering for planar beam splitters

Planar mirror beam-splitters (semi-transparent mirrors) enable us to optically combine stereoscopically projected 3D graphics with the real environment and to perceive both environments (real and virtual) in conjunction:

A planar mirror beam-splitter divides the environment into two subspaces (cf. figure 4.5)

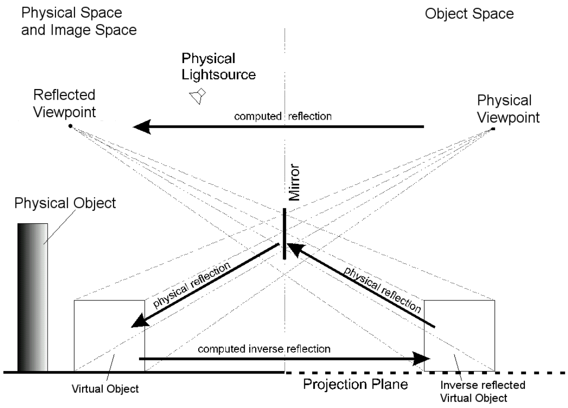


Figure 4.5: *Perspective and geometric symmetry of the reflected model-view transform.* [8]

We call the subspace that contains the physical viewpoint and the projection plane (optically: real objects in form of illuminated pixels on the projection screen) the *object space* and the subspace that contains the physical objects and additional physical light sources the *physical space*. Note that from a geometric optics point of view, the physical space equals (or overlays) the mirror's *image space* (i.e., the space that appears behind the mirror by looking at it).

As in the immersive case, virtual objects that consist of graphical elements (such as geometry, normals, textures, clipping planes, virtual light sources, etc.) are defined within the 3D freespace (i.e., within the global coordinate system of the virtual environment). In our case, this coordinate system actually exceeds the boundaries of the projection space and extends into the surrounding physical space.

The challenge is to complement the physical objects located within the physical space with additional virtual augmentations. To achieve this, the graphical elements of the virtual objects are either defined directly within the physical space, or they are transformed to it during an object registration process. In both cases, graphical elements are virtually located within the physical space and, due to the lack of projection possibilities in the physical space, are not visible to the observer without additional aids.

4.2.1.1. Reflection transformation

We now consider the mirror and compute the reflection of the viewer's physical eye locations (as well as possible virtual headlights). We then apply the inverse reflection to every graphical element that is located within the physical space. In this manner these graphical elements are transformed and can be projected at their corresponding inverse reflected position within the object space. Thus, they are physically reflected back by the mirror into the mirror's image space. We refer to this as the reflected model-view transform [8].

When the setup is sufficiently calibrated, the physical space and the image space overlay exactly. The graphical elements appear in the same position within the image space as they would within the physical space without the mirror (if a projection possibility was given within the physical space).

With known plane parameters of the beam splitter within the world coordinate system, a point in 3D space can be reflected with:

$$P' = P - 2(NP + d)N \qquad (4.1)$$

Where P' is the reflection of P over the mirror plane $[N,d] = [a,b,c,d]$.

This is equivalent to multiplying P with the 4x4 reflection matrix:

$$R = \begin{bmatrix} 1-2a^2 & -2ab & -2ac & -2ad \\ -2ab & 1-2b^2 & -2bc & -2bd \\ -2ac & -2bc & 1-2c^2 & -2cd \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad (4.2)$$

Note that $R = R^{-1}$.

The reflected viewpoint can be computed with equation 4.1 or by multiplying the physical (tracked) viewpoint with the reflection matrix defined in equation 4.2.

The inverse reflection of a virtual object that is located within the physical space is simply computed from its reflection with respect to the mirror plane. Since we assume that the physical space and the mirror's image space exactly overlay, we can also assume that the reflection of the graphical elements located within the physical space results in the inverse reflection of the image space, that is, they are transformed to their corresponding positions within the object space and can be displayed on the projection plane. Consequently, the additional model transformation (i.e., the inverse reflection of the scene) is achieved by multiplying the reflection matrix (equation 4.2) onto the current model-view matrix of the transformation pipeline (between scene transformation and view transformation).

If we properly illuminate the physical objects located within the physical space, the mirror beam-splitter transmits their images. However, it also physically reflects the image of the inverse reflected graphical elements, which is projected within the object space. Thus both the transmitted image of the real objects and the reflection of the displayed graphics are simultaneously visible to the observer by looking at the mirror.

By applying the reflection matrix, every graphical element is reflected with respect to the mirror plane. A side effect of this is that the order of reflected polygons is also reversed (e.g., from counterclockwise to clockwise) which, due to the wrong front-face determination, results in a wrong rendering (e.g., lighting, culling, etc.). This can easily be solved by explicitly reversing the polygon order. Note that transformations and rendering have to be done for both viewpoints (left and right) if stereoscopic rendering is activated.

4.2.1.2 Refraction transformation

Mirror beam-splitters that are built from thick plates of glass (such as the Extended Virtual Table) cause optical distortion that result from refraction.

With respect to figure 4.4: All scene vertices that are registered to the physical space are virtual points that are not physically located behind the mirror, and consequently are not physically refracted by the glass, but are reflected by the front surface mirror. Since the transmitted light which is emitted by the physical objects and perceived by looking through the mirror is refracted, but the light that is reflected by the front surface mirror is not, the transmitted image of the physical space cannot be precisely registered to the image space (i.e., the reflected object space), even if their geometry and alignment match exactly within our world coordinate system. Unfortunately refraction cannot be undistorted by an affine transformation, but approximations exist that are sufficient for the described AR display types.

All optical systems that use any kind of see-through element have to deal with similar problems. While for HMDs, aberrations caused by refraction of the lenses are mostly assumed to be static[*] (as stated by Azuma [3]), they

[*] We do not consider rotations of the eyeballs.

can be corrected with paraxial analysis approaches. For other setups, such as the reach-in systems that were mentioned in chapter 2 or other spatially aligned mirror beam-splitters, aberrations caused by refraction are dynamic, since the optical distortion changes with a moving viewpoint. Wiegand et al [129], for instance, estimated the displacement caused by refraction for their setup to be less than 1.5 mm, predominantly in +y-direction of their coordinate system. While an estimation of a constant refraction might be sufficient for their apparatus (i.e., a near-field virtual environment system with a fixed viewpoint that applies a relatively thin (3 mm) mirror), some spatial optical see-through setups require a more precise definition, because they are not near-field VE systems but rather mid-field VR/ AR systems, they consider a head-tracked viewpoint, and they might apply a relatively thick beam-splitter.

Since we cannot pre-distort the refracted transmitted image of the physical space, we artificially refract the virtual scene within the image space instead, in order to make both images match. However, since refraction is a complex curvilinear transformation and does not yield a stigmatic mapping in any case, we can only approximate it.

A simple solution is the assumption that, against its optical nature, refraction can be expressed as an affine transformation. For beam-tracing, Heckbert [44] assumes that, considering only paraxial rays, objects seen through a polygon with the refraction index appear to be times their actual distance. For this approximation, he does not take the incidence angles of the optical line of sight into account but, instead, assumes a constant incidence angle of that is define by the optical axis which is perpendicular (on-axis) to the refracting polygon.

In [11] we present an affine approximation that can be applied to pre-distort refraction in off-axis situations:

$$\Delta = T\left(1 - \frac{\tan\beta}{\tan\alpha}\right) \qquad (4.3)$$

where T is the thickness of the beam splitter, α the angle between the plane normal of the mirror and the line of sight, and β is given by Snellius' law of refraction:

$$\eta_1 \sin\alpha = \eta_2 \sin\beta \qquad (4.4)$$

In case of a beam splitter that is located in air, the refraction index η_1 is equal to one. The refraction index η_2 is the one of the mirrors' base material (e.g., glass). Since we assume refraction is equivalent to the geometric transformation of the refracted object along the normal of the beam splitter, it can be expressed as a 4x4 transformation matrix:

$$F = \begin{bmatrix} 1 & 0 & 0 & \Delta a \\ 0 & 1 & 0 & \Delta b \\ 0 & 0 & 1 & \Delta c \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad (4.5)$$

In this case, the incidence angle of a dynamically changing line of sight is taken into account, rather than considering only paraxial rays. The viewing direction of the observer is determined by a head-tracker.

Note that several other approximations are presented in [11, 14]. However, they are based on a more precise curvilinear refraction transformation and cannot be integrated into standard transformation pipelines of today's graphics hardware.

4.2.1.3 Single user, single screen and single beam splitter

The algorithm below summarizes the rendering steps for spatial optical see-through displays that apply one planar screen and one planar beam splitter (such as the Extended Virtual Table – section 4.1.1). Only one user is supported.

```
for left and right viewpoints $i$
        initialize transformation pipeline and polygon order
        compute reflected viewpoint $v_i' = Rv_i$
        compute refraction offset $\Delta_i$ for $i$
        set transformation pipeline: $RMF_iV_iP$
        reverse polygon order
        render scene from $v_i'$
endfor
```

First, the polygon order and the transformation pipeline have to be set to an initial state. Then the reflected viewpoint and the view-dependent refraction offset (Δ_i) are computed. The transformation pipeline is then set to the corresponding concatenation of transformation matrices: reflection transformation (R), model transformation (M), refraction transformation (F_i, with v_i), view transformation (V_i, with v_i') and the projection transformation (P). Finally, the polygon order is reversed and the scene is rendered. Note that we assume that the screen coordinate system is equivalent to the world coordinate system. Note also that R might not be static but has to be re-computed continuously (e.g., if moving configurations such as the Transflective Pad (cf. 2.19) have to be supported).

4.2.1.4 Single user, single screen and multiple beam splitters

To support configurations that apply multiple planar beam splitters and a single screen (such as the pyramid-shaped Virtual Showcase variation shown in figure 4.3-bottom) the following algorithm can be applied for a single user:

```
for left and right viewpoints $i$
    for each to $i$ front-facing beam splitter $j$
        initialize transformation pipeline and polygon order
        compute reflected viewpoint $v_i' = R_j v_i$
        compute refraction offset $\Delta_{ij}$ for $i$ and $j$
        set transformation pipeline: $R_j M F_{ij} V_i P$
        reverse polygon order
        render scene from $v_i'$
    endfor
endfor
```

The scene has to be rendered for each viewpoint and each beam splitter. If the configuration of the optics unequivocally tessellates 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 (cf. figure 4.3-bottom).

4.2.1.5 Multiple users, single screen and multiple beam splitters

As described in section 4.1.2.2, multiple users can be supported simultaneously by assigning them to an individual beam splitter and screen portion (we referred to this technique as screen-partitioning). The following algorithm will support such kind of configurations (an example is illustrated in figure 4.3-upper left/bottom):

```
for left and right viewpoints $i$ of all viewers
    initialize transformation pipeline and polygon order
    compute reflected viewpoint $v_i' = R_i v_i$
    compute refraction offset $\Delta_i$ for $i$
    set transformation pipeline: $R_i M F_i V_i P$
    reverse polygon order
    render scene from $v_i'$
endfor
```

Note that each user is restricted to the viewing zone that is given by the assigned beam splitter.

4.2.1.6 Multiple users, multiple screens and multiple beam splitters

Display configurations that assign an individual screen and beam splitter to each user benefit from a higher resolution those configurations that split the screen (and its resolution) into different portions. An example is the pyramid-shaped Virtual Showcase variation illustrated in the appendix 1. Here the resolution per user is increase by factor 5, compared to the prototype shown in figure 4.3. To support such configurations, an additional screen transformation (S) has to be introduces. This matrix represents a concatenation of several translations, rotations and scaling transformations, that map coordinates from the world coordinate system into an individual coordinate system of the corresponding screen. One can also think of the inverse transformation of the screen itself within the world coordinate system.

The following algorithm can be applied in combination with such configurations:

```
for left and right viewpoints $i$ of all viewers
        initialize transformation pipeline and polygon order
        compute reflected viewpoint $v_i' = R_i v_i$
        compute refraction offset $\Delta_i$ for $i$
        set transformation pipeline: $R_i M F_i V_i S_i P$
        reverse polygon order
        render scene from $v_i'$
endfor
```

Note that by keeping track of individual model transformations (M) and by rendering individual scenes for each user, the display configuration practically acts as multiple user-individual displays. Otherwise, multiple users share and interact with the same scene. To use individual model transformations but to render the same scene for each user has been proven to be an effective way of precisely registering virtual objects to real ones – cancelling out slight physical misalignments of the screens and mirrors (whose measured/calculated parameters that are stored in R_i and S_i reflect these errors) within each M. Thereby, the object transformation that is used to align real and virtual objects is stored in each M. This also applies for the algorithm that is presented in section 4.2.1.5 – although only one screen is used.

4.2.2. Multi-pass rendering for curved beam splitters

If the single elements of such assemblies, described in the section above are made infinitely small, we can say that the optics is curved, rather than built from multiple planar sections. Approximating an appropriate warping of the image with a finite number of affine transformations, however, is inefficient for curved optics.

For the cone-like Virtual Showcase (cf. figure 2.11) an image-based two-pass rendering method has been presented in [10], which warps a generated image by projecting it individually for each pixel (c.f. figure 4.6).

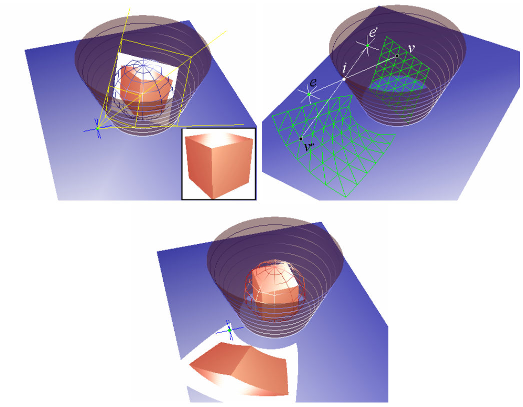


Figure 4.6: *Two-pass rendering for curved mirror beam splitters: generation (upper-left), warping (upper-right), and displaying (bottom) of image.* [10]

First, an image of the showcase's virtual content is generated from the observer's point of view using a traditional on-axis perspective projection (c.f. figure 4.6-upper left). The frustum of the projection is defined by the viewpoint and the bounding sphere of the scene. This image is geometrically approximated by a simple uniformly tessellated grid that is transformed into the current viewing frustum in such a way that it is positioned perpendicular to the optical axis. Next, the grid vertices are transformed with respect to the viewpoint and the mirror optics and projected onto the display surface (c.f. figure 4.6-upper right). Finally, the image that was generated during the first pass is mapped onto the warped grid using texture mapping and bi- or tri-linear texture filtering during the second pass (c.f. figure 4.6-bottom). This process is repeated for multiple individual viewpoints when stereoscopic viewing and/or multiple observers need to be supported.

Note that we have developed advanced variation of this image warping method:

In [16] we describe a quadtree-based selective refinement algorithm. The algorithm also applies two-pass rendering, and recursively generates and selectively refines the underlying image grid with locally adapted levels of detail (instead of a uniform grid). Each discrete level of detail is determined by computing and evaluating the displacement error on the image plane within the image space of the mirror optics. The algorithm generates and refines only those image portions that contain visible information by considering an oriented convex container of the rendered scene. While display specific details of the algorithm are explained based on the more complex case of a particular mirror display, the Virtual Showcase its general functionality is valid for other non-linear displays.

A further enhancement is the progressive version [14] of this algorithm which dynamically adapts the error thresholds that trigger the grid refinement to reach and keep a desired frame rate. This method also considers the texture resolution to further accelerate our multi-pass rendering method.

It is beyond the scope of this tutorial to describe these enhancements in detail. The interested reader is referred to the referenced publications. The basic warping algorithm as well as the enhancements, however, apply the same transformations that correct reflection and refraction distortions introduced by the beam splitter optics. These will be discussed within the sections below.

Note that most of the following transformations support a curvilinear image deformation. Since this is achieved by per-vertex operations, they are not supported by standard transformation pipelines. Consequently they (i.e., matrix operations and transformations) have to be implemented in software until hardware accelerated procedural shaders are powerful enough to support them.

4.2.2.1. Reflection transformation

For all grid vertices that are geometrically located inside the mirror optics, the intersection of the geometric line of sight (i.e., the ray that is spanned by the eye and the vertex) with the mirror geometry is computed first.

An efficient way of describing conical mirror optics, for instance, is to apply an explicit function. This function can be used to calculate the intersections and the normal vectors (using its 1st order derivatives):

$$f(x,y,z)=\frac{x^2}{a^2}+\frac{y^2}{b^2}-\frac{z^2}{c^2}=0,$$

$$N(x,y,z)=2\left[\frac{x}{a^2},\frac{y}{b^2},-\frac{z}{c^2}\right] \qquad (4.6)$$

where a,b are the cone's radii with its center located at the world-coordinate-system's origin, and c is the cone's height along the z-axis.

After a geometric line of sight has been transformed from the world-coordinate-system into the cone-coordinate-system, it can be intersected easily with the cone by solving a quadratic equation created by inserting a parametric ray representation into the cone equation.

Note that not all shapes can be expressed by explicit functions. In such cases the geometry of the optics can be approximated by a high-resolution triangle mesh and appropriate ray intersection methods have to be applied. Details are described in [14].

Next, the normal vector at the intersection has to be determined. The intersection point, together with the normal vector, gives the tangential plane at the intersection. Thus, they deliver the plane parameters for the per-vertex reflection transformation.

Then equation 4.1 is applied to reflect the grid vertex as well as the viewpoint over the computed tangent plane.

Finally, the reflected grid vertex has to be projected onto the screen plane from the reflected viewpoint.

Given a projection origin (the reflected viewpoint in our case) and parameters of the screen plane, the following 4x4 transformation matrix can be used to project a 3D vertex (the reflected grid vertex in our case) onto an arbitrary plane (the screen plane in our case):

$$P=\begin{bmatrix} k-a_p x & -b_p x & -c_p x & -d_p x \\ -a_p y & k-b_p y & -c_p y & -d_p y \\ -a_p z & -b_p z & k-c_p z & -d_p z \\ -a_p & -b_p & -c_p & -d_p \end{bmatrix} \qquad (4.7)$$

where a_p,b_p,c_p,d_p are parameters of the projection plane, x,y,z are the coordinates of the projection center (i.e., the reflected viewpoint), and $k=\lfloor a_p,b_p,c_p,d_p\rfloor\cdot[x,y,z,1]$.

Note that since P is a perspective projection, a perspective division has to be done accordingly to produce correct device coordinates.

4.2.2.2. Refraction transformation

The following section describes how to apply refraction transformation to the image that has been generated during the first rendering pass (cf. figure 4.6). Note that the image is refracted before the image-based reflected model-view transformation is applied to the image geometry.

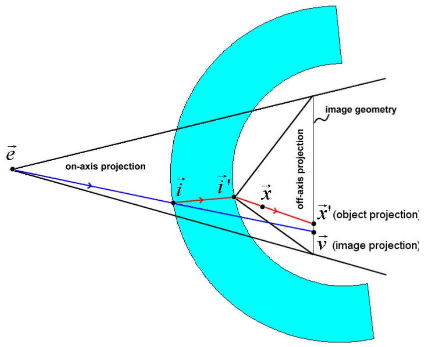


Figure 4.7: *Refracted image transformation for curved beam splitters: Geometric lines of sight (blue) through mediums with different densities are refracted. The corresponding optical lines of sight (red) can be computed with Snell's law of refraction.* [14]

Given an image (grid) vertex $\vec{v}$ and the viewpoint $\vec{e}$, the according geometric line of sight is computed. Using Snell's law of refraction (eqn. 4.4) together with the vectorized equation for specular refraction rays,

$$R' = \frac{\eta_1}{\eta_2} R - \left(\cos\beta + \frac{\eta_1}{\eta_2} NR \right) N \qquad (4.8)$$

the corresponding optical line of sight can be determined by computing the in/out refractors at the associated surface intersections.

We could now determine the refraction of the image vertex $\vec{v}$ by computing the geometric intersection of the out-refractors with the image geometry. To simulate refraction, however, we only need to ensure that the pixel at $\vec{x}'$ (i.e., the projection of an object) will be seen at the location of $\vec{v}$ (i.e., the projection of the image at which the corresponding object optically appears). Instead of generating a new image vertex at $\vec{x}'$ and transforming it to the location of $\vec{v}$, we can also assign the texture coordinate at $\vec{x}'$ to the existing vertex $\vec{v}$.

In this case, we can keep the number of image vertices (and consequently the time required for the reflection transformation) constant. The intersection of the in-refractor with the outer mirror surface $\vec{i}'$ and an arbitrary point on the out-refractor $\vec{x}$ are transformed into the coordinate system of the image geometry, next.

The composition of an appropriate texture matrix that computes new texture coordinates for the image vertex is outlined in the algorithm below:

```
compute texture normalization correction*:
    $s = scale(0.5,0.5,0.5), s = s \cdot translate(1,1,0)$
compute off-axis projection transformation:
    $\phi = \frac{(\vec{i}'_z - 1)}{\vec{i}'_z}$, $left = -\phi(1+\vec{i}'_x)$, $right = \phi(1-\vec{i}'_x)$
    $bottom = -\phi(1+\vec{i}'_y)$, $top = \phi(1+\vec{i}'_y)$
    $near = \vec{i}'_z - 1$, $far = \vec{i}'_z + 1$
    $p = frustum(left, right, bottom, top, near, far)$
compute view transformation:
    $v = lookat(\vec{i}'_x, \vec{i}'_y, \vec{i}'_z, \vec{i}'_x, \vec{i}'_y, 0,0,1,0)$
compute new texture coordinate $\vec{x}'$ for particular $\vec{x}(\vec{v})$,
including perspective division
    $\vec{x}' = S \cdot \frac{(P \cdot V \cdot \vec{x})}{w}$
```

As illustrated in figure 4.7, an off-axis projection transformation is applied, where the center of projection is $\vec{i}'$. Multiplying $\vec{x}$ by the resulting texture matrix and performing the perspective division projects $\vec{x}$ to the correct location within the normalized texture space of the image. Finally, the resulting texture coordinate $\vec{x}'$ has to be assigned to v'.

Note that if the image size is smaller than the size of the allocated texture memory, this difference has to be considered for the normalization correction. In this case, the image's texture coordinates are not bound by the value range of [0,1].

Note that the matrix operations can be replaced by an explicit ray-casting.

*This applies for OpenGL-like definitions of the texture and normalized device coordinates.

Note also that the derivation of the optical lines of sight for planar lenses is less complex, since in this case the optical lines of sight equal the parallel shifted geometric counterparts (cf. figure 4.8).

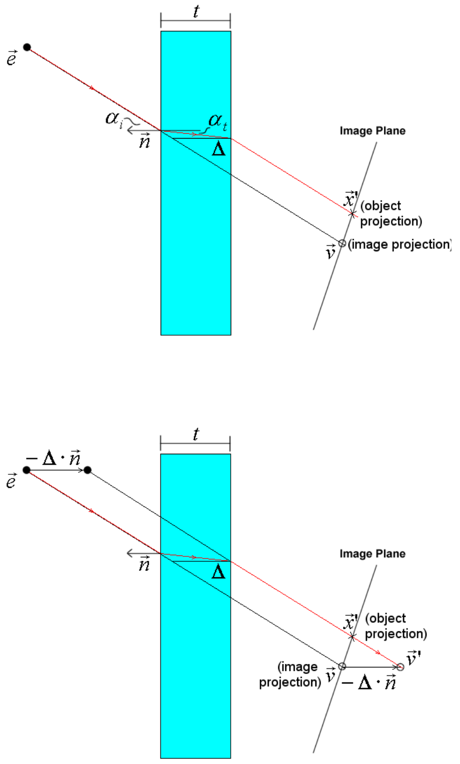


Figure 4.8: *Refracted image transform for planar beam splitters.* [14]

In this case, the inner and outer surfaces share the same normal vector, but are offset by the thickness t. Consequently, the optical line of sight does not have to be traced through the lens to determine the entrance vector and normal at the exit point $\vec{i}'$. Instead, the entrance and exit angles (here denoted as $\alpha_i = \alpha, \alpha_t = \beta$), have to be computed (using Snell's law, eqn. 4.4) for the given geometric line of sight, normal vector and refraction index. Then, the transformation offset Δ can be computed using equation 4.3. The texture matrix is composed in a similar way as outlined by algorithm above. However, due to the refraction properties of planar lenses, the center of projection is given by $\vec{e}'-\Delta\vec{n}$ (instead of using $\vec{i}'$). To determine the correct texture coordinate $\vec{x}'$ for a given image vertex $\vec{v}$, $\vec{v}'=\vec{v}-\Delta\vec{n}$ has to be projected onto the image plane.

Note that the matrix given by equation 4.7 can be used instead of the matrix that results from a projection-model-view composition as implemented with OpenGL's glFrustum and gluLookAt (Note that again that these per-vertex computations are not supported by standard transformation piplelines. We refere to the OpenGL implementations only to give an example. Matrices operations and transformations have to be re-implemented in software). However, a different texture normalization is required, since the matrix in equation 4.7 delivers world coordinates, rather than normalized device coordinates.

Nevertheless, both refraction methods face the following problems for outer areas on the image:

- Given a geometric line of sight to an outer image vertex, its corresponding optical line of sight does not intersect the image. Thus, an image vertex exists but its new texture coordinate cannot be computed. This results in vertices with no, or wrong texture information;
- Given an optical line of sight to an outer pixel on the image, its corresponding geometric line of sight does not intersect the image. Thus, a texture coordinate can be found but an assignable image vertex does not exist. Consequently, the portion surrounding this pixel cannot be transformed. This results in image portions that aren't mapped onto the image geometry.

A simple solution to address these problems does not avoid them, but ensures that they do not occur for image portions which contain visible information: The image size depends on the radius of the scene's bounding sphere (see section 4.2.2). We can simply increase the image by adding some constant amount to the bounding sphere's radius before carrying out the first rendering pass. An enlarged image does not affect the image content, but subjoins additional outer image space that does not contain any visible information (i.e., just black pixels). In this way, we ensure that the above mentioned problems emerge only at the new (black) regions. Yet, these regions will not be visible as reflections in the mirror.

Note that the refracted image transformation represents another transformation of the image generated during the first rendering pass. In contrast to the image-based reflected model-view transformation (section 4.2.2.1) which transforms image vertices, the refracted image transform re-maps texture coordinates. However, all image transformations have to be applied before the final image is displayed during the second rendering pass.

Optically, the refracted image transformation determines projections of objects $\bar{x}'$ for discrete image projections $\bar{v}'$, and interpolates object projections for intermediate image projections by making use of texture mapping. In this case, image projections and corresponding object projections can be found at different pixel locations within the image.

4.3. Projector-based illumination

Instead of using normal light sources (such as simple light bulbs) to illuminate real objects, video projectors can be applied. Video projectors support a per-pixel illumination as well as a synchronization of rendering the graphical overlays and illuminating the real objects.

The sections below describe techniques that apply a projector based illumination to create consistent occlusion and lighting effects for mixed scenes displayed with spatial AR setups, such as the Virtual Showcase.

How to combine the effects of multiple projectors is described in section 3.4.

4.3.1. Realistic occlusion effects

One of the main challenges for optical see-through AR displays – spatial or head-attached is the generation of correct occlusion effects between virtual and real objects. If normal light bulbs are used to illuminate the real scene, light that is diffused off the real surfaces interferences with the light (i.e., the graphics) that is reflected by the optical combiner. The result is that graphical overlays appear semi-transparent.

A solution for this problem that supports head-attached displays, ELMO [58], has been illustrated in figure 2.9.

For spatial optical see-through displays, such as the Virtual Showcase, we have introduced a method that uses video projectors (called *light projectors*) instead of simple light bulbs to illuminate the real scenery [12].

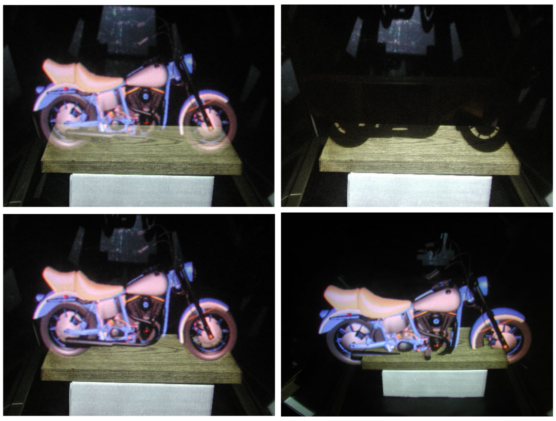


Figure 4.9: *Wrong occlusion effects with normal illumination (upper-left). Occlusion shadow generated with projector-based illumination (upper-right). Realistic occlusion of the real object by the virtual one (lower-left). Knowing depth information of real objects allows the occlusion of virtual objects by real ones (lower-right).* [12]

We dynamically generate shadows (called *occlusion shadows*) directly on the real objects' surfaces wherever graphics is overlaid. These shadows are not directly visible to the observers, since they are purposely occluded by the overlaid graphics. We additionally render phantom bodies representing real objects which occlude virtual objects behind them. The combination of occlusion shadows and phantom bodies effectively solves the occlusion problem for spatial optical see-through displays such as the Virtual Showcase.

For rendering occlusion shadows the viewpoints of each user, the intrinsic and extrinsic parameters of each light projector, as well as the virtual and the real scene must be known.

The viewpoints are continuously measured with head-tracking technology, while the light projectors' parameters are determined only once during a calibration phase. Virtual objects can be interactively manipulated within the showcase during runtime.

Knowing the scene and the view transformation lets us compute the perspective projection matrix (V) of the corresponding viewpoint that incorporates the model-view transformation with respect to the scene's origin.

The light projectors can be calibrated to a registered geometric representation of the real scene that is registered to its physical counterpart. To do this we apply a semi-manual calibration routine: We interactively mark the two-dimensional projections of known three-dimensional fiducials on a projector's image plane. Using these mappings, we apply numerical minimization method to solve a perspective n-point problem. This results in the perspective 4x4 projection matrices P of the projector that incorporates the correct model-view transformation with respect to the scene origin. More details are presented in [12].

If multiple projectors are used, the calibration process has to be repeated for each projector separately.

The basic algorithm below illustrates how to render occlusion shadows for a single point of view. The depth information of both – the real and the virtual content have to be known. A shadow mask that contains the silhouette of the virtual content is generated which is then perspectively mapped onto the known geometry of the real content. Then we render the illumination for the real content into the frame buffer. For now we want to assume that we project only uniformly colored light onto the real surfaces from the light projector's point of view while virtual objects are illuminated from the positions of the virtual light sources. This illumination, however, could be computed with a more an advanced BRDF model – producing a correct and matching radiance on real and virtual surfaces with respect to virtual light sources. We will describe this in section 4.3.2.

```
generate shadow mask:
    set projection matrix to $V$
    render real content into depth buffer
    render virtual content into stencil buffer
    render illumination for real content into …
    …frame buffer (previously cleared to black)

read-back:
    transfer frame buffer into texture memory $T$

render shadow mask:
    set projection matrix to $P$
    set texture matrix to $V$ + normalization space
    correction
    clear frame buffer to black
    render real content into frame buffer using …
    …projective texture $T$
```

Note also that to render the real content into the depth buffer ensures a correct occlusion of virtual objects by real ones. The normalization space correction consists of a scaling by [0.5,0.5,1.0], followed by a translation of [0.5,0.5,0.5] to map from normalized screen space to normalized texture space*.

A clear limitation of this basic method is the following fact: If the same real surfaces are simultaneously visible from multiple points of view (e.g. for different observers), individual occlusion shadows that project onto these surfaces are also visible from different viewpoints at the same time.

Considering two observers, for instance, observer A might be able to see the occlusion shadow that is generated for observer B and vice versa. In addition, the shadows move if the viewers are moving, which might be confusing.

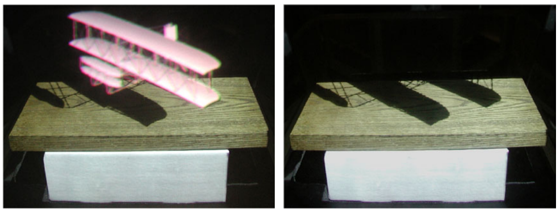


Figure 4.10: *Occlusion shadows generated for two different viewpoints. With graphical overlay (left), and without graphical overlay (right).* [12]

We have proposed several methods to reduce or avoid these effects:

Approach one [12]: Occlusion shadows generated for other viewpoints are the umbral hard-shadows that are cast by the virtual scene with a light source positioned at the other viewpoints' locations. We can make use of this fact by attaching a point light to each viewpoint. This generates correct lighting effects on the virtual scene's surfaces – in addition to matching hard-shadows on the real scene's surfaces.

Approach two [12]: We try to minimize the interference between individual occlusion shadows by ensuring that they are generated only on those real surfaces that are visible from the corresponding viewpoint. However, since the occlusion shadows are finally rendered from the viewpoint of the projector, all view-dependent computations (e.g., back-face culling and depth buffering)

*This applies for OpenGL-like definitions of the texture and normalized device coordinates.

are done for this perspective – not for the perspectives of the actual viewpoints.

Approach three [17]: We complement each light projector by a video camera (we call this combination *responsive light projector*) that is used to dynamically scan reflectance information from the surfaces of the real objects. Knowing this reflectance information, however, leads to an effective and general solution: In addition to the virtual scene, we render the portions of the real scene (i.e., its registered reflectance map) that are covered by the occlusion shadows of all other observers. If this is done well, we can create seamless transitions between the real and the virtual portions. For each observer the occlusion shadows of all other observers are rendered into the stencil buffer first. This is done by rendering the real scene's geometry from each observer's perspective and adding the corresponding occlusion shadows via projective texture mapping. The stencil buffer has to be filled in such a way that the area surrounding the occlusion shadows will be blanked out in the final image. Then the real scene's reflectance map is rendered into the frame buffer (also from the perspective of the observer) and is shaded under the virtual lighting situation. After stenciling has been disabled, the virtual objects can be added to the observer's view.

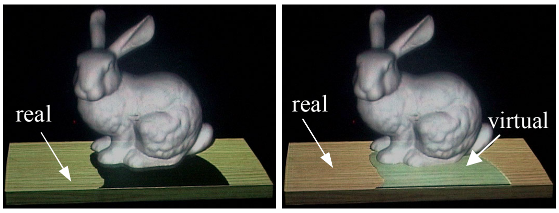


Figure 4.11: *Occlusion shadow of second observer is clearly visible (left). Wrongly visible occlusion shadow is covered by optically overlaying the corresponding part of the reflectance map (right).* [17]

Due to self occlusion, not all portions of the real content can be lit by a single light projector. A solution to this problem is to increase the number of projectors and place them in such a way that the projected light is distributed over the real content. To guarantee a uniform illumination, however, surfaces should not be lit by more than one projector at the same time or with the same intensity. Otherwise, the projected light accumulates on these surfaces and they appear brighter than others. In [12] we describe a method that subdivides the geometry of the real content into surface portions that are assigned to, and finally rendered by an individual light projector.

The crossfeathering method presented in section 3.4, however, is a better choice for merging the images of multiple projectors on a pixel-basis, which could also be used with our setup.

4.3.2. Consistent illumination

To achieve a consistent lighting situation between real and virtual environments is important for convincing augmented reality applications. Matching direct and indirect lighting effects, such as shading, shadows, reflections and color bleeding can be approximated in such a controlled mixed environment [17].

The basic idea is to first neutralize the physical illumination effect caused by a light projector and then use it to simulate virtual light sources. To create consistent shading and shadow effects on both – real and virtual objects – and to benefit from hardware acceleration, we perform the following algorithm:

```
create an intensity image $I_{\text{int}}$ of the real object:
        render real object from perspective of light projector…
        …having a white diffuse material…
        …illuminated by a white virtual point light source…
        …located at the projector
create a shading image $I_{shad}$ of real and virtual objects:
        generate shadow map for real and virtual objects
        render real objects from perspective of light projector…
        …having a white diffuse material…
        …illuminated by the virtual light sources…
        …with shadow mapping enabled
compute irradiance image $I_{irrad} = \frac{I_{shad}}{I_{\text{int}}}$
render occlusion shadows from perspective of projector…
…and blend with $I_{irrad}$
```

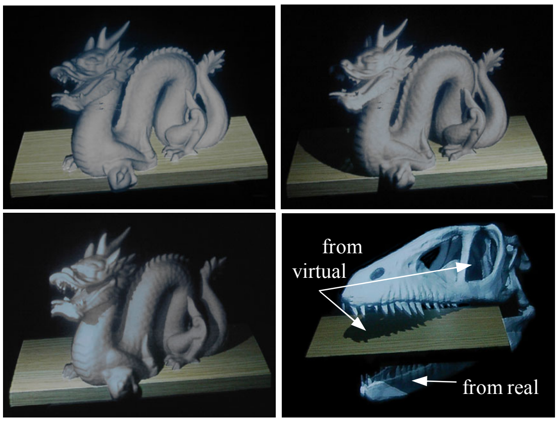


Figure 4.12: *Unrealistic illumination with occlusion shadows (upper-left). Realistic illumination under varying virtual lighting conditions with matching shading and shadows (upper-right, lower-left and -right).* [17]

For more details on creating consistent illumination effects with responsive light projectors (also direct reflections and indirect color bleeding), the interested reader is referred to [17].

5. Current areas of application

5.1. Scientific visualization and digital storytelling within a museum context

Museums are facing more and more competition from theme parks and other entertainment institutions. Some museums, however, have learned that their visitors do not only want to be educated, but also be entertained. Consequently they are investigating new *edutainment* approaches. Technologies such as multimedia, virtual reality and augmented reality in combination with appropriate interaction and storytelling techniques may become the new presentation forms of future museums.

In this section we discuss two concrete case studies within the palaeontology domain for which spatial augmented reality has been used to provide scientific visualization capabilities and –in combination with digital storytelling techniques– to support the presentation of scientific findings to a non-scientific audience. The acceptance of such new presentation forms has been evaluated for one of the case studies. Our results will be presented below.

5.1.1. Scientific visualization for palaeontology

Some of the questions that paleontologists try to answer are, what did dinosaurs look like, how did they breath, smell and eat. Only the fossilized bones and teeth –the hard parts– of dinosaurs are generally preserved, but the soft tissues that clothe and animate the bones are the key to unlocking the secrets of dinosaur biology. Witmer and his colleagues seek to remedy this situation by reconstructing such soft-tissue components as muscles, veins, arteries, or cartilage [130]. Soft-tissues carve distinct marks in the bone's structure that can be compared to the marks found on bones of modern-day animals, such as crocodiles and birds – the closest living relatives of dinosaurs.

With the aid of artists and sculptors, the conceptual image that has been drawn in the paleontologists' minds can finally be visualized. Two-dimensional drawings (e.g., sketched over photographs of the bones), or three-dimensional clay or plastic models are sometimes used to illustrate how soft-tissue structures share the limited space within complex anatomic areas such as the skull. These forms of expression are also commonly used in museum displays. The main drawback of these methods for paleontologists and museum visitors is their lack of dynamics and interactivity. Drawings and sculptures are static and difficult to modify after they have been created.

Augmented reality holds the prospect of allowing us to visualize reconstructed soft tissues with the physical bone structure [13]. For paleontologists, this offers the opportunity to test models of soft-tissue anatomy by assessing the conformational relationships of reconstructed components. The limited space within the skull, for instance, is shared by numerous components, ranging from eyeballs and tongues to jaw muscles and nasal cartilages. Augmented reality allows paleontologists to directly assess how these diverse anatomical elements are "packed" within the skull. Moreover, it allows us to study how soft-tissues perform in accomplishing biological tasks, such as the muscle contraction of working jaws and how the bulging of those contracting muscles dynamically affect the conformation of surrounding structures. For visitors of museums, augmented reality in combination with digital story telling enables a more exciting and interactive museum experience, and a potentially improved knowledge transfer (see section 5.1.2).

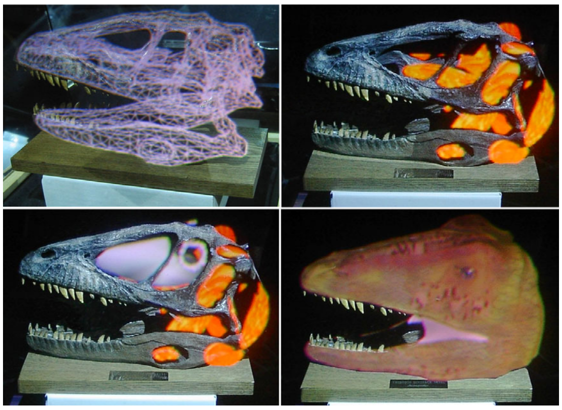


Figure 5.1: *Visualization with the Virtual Showcase: The physical skull of Deinonychus is placed inside the display. A scanned skull geometry is registered to the real counterpart (upper-left). Different muscle groups are augmented (upper-right). The paranasal air sinuses and the bony eye rings are integrated into the skull (lower-left). The skin is superimposed on the skull (lower-right).* [13]

In the following, we present an example of how a cast skull of an Early Cretaceous dinosaur, a *Deinonychus*, has been augmented with 3-D computer models of reconstructed soft tissues and missing bones. The real and virtual components can be perceived together in stereo from any perspective using the Virtual Showcase display.

Several data sets of the bones and the soft-tissues have been acquired from different sources. The skull bones, for example, have been captured using 3D laser-scanning technology. Sets of reconstructed muscles, eyeball, nostrils, ears, sinuses, and skin were assembled by Witmer's lab and have been pre-modeled with an off-the-shelf modeling tool.

To visualize the packing process (cf. figure 5.1) and to present the final results with the Virtual Showcase, the physical skull is placed inside the display. The scanned geometric representation of the skull is then registered to its physical counterpart. It will later serve to compute illumination and occlusion effects directly on the skull's surface. The pre-modeled soft-tissues can then be interactively placed inside the skull with a simple mouse-based interface allowing us to investigate contact points and collisions with the physical bones and other soft-tissues.

The virtual components appear within the same 3-D space as the skull bones. This is achieved via a stereoscopic presentation of the graphics. Using head-tracking technology, multiple users are able to walk around the display – observing the augmented artifact from different perspectives.

Simulating realistic occlusion effects between the physical bones and the virtual soft-tissues is essential for packing and presentation. The techniques described in section 4.3.1. have been applied to solve this problem.

The video projectors are used to create view-dependent lighting effects on the real skull's surface. Generating shadows on the physical object exactly underneath the overlaid graphics, for instance, allows virtual parts to mutually occlude the underlying real surfaces. Having depth information of the skull on the other hand, allows us to cull the occluded graphics before they are displayed. Thus, the physical bones can occlude virtual components and vice versa. This strongly enhances the interactive packing process, as well as the realism of the presentation.

To simulate interaction and behavior in different situations the virtual components are animated during packing and presentation. To render the graphics, a conventional game engine that provides both high-quality animation and interactive frame rates is used. Enhanced by synchronized audio output and projector-based illumination, which allows us to dynamically fade in and out specific parts of the physical skull, an effective form of digital story telling can be achieved. Different soft-tissue layers and components are displayed over time while their relation and functions are explained with a variety of multi-media aids, such as voice, text annotations, graphical animation and lighting effects.

Gatesy et al. [41] are analyzing 210-million-year-old fossilized dinosaur footprints that they discovered on rocky exposures in eastern Greenland. Their research focuses on how theropods, (bipdal carnivorous dinosaurs) moved about on two legs and how this locomotor pattern evolved over time. Gatesy and his colleagues are especially interested in using footprints to discern similarities and differences among early theropods and birds, their living descendants. Shallow footprints made on firm mud record the shape of the bottom of the foot but contain very little information about how the limb moved while on the ground. However, many of the Greenlandic trackways were made by theropods that sank to varying depths in soft mud. Such deep prints record the path of the foot through a volume of sediment in three-dimensions, thereby allowing limb motion to be reconstructed.

This dynamic perspective has helped yield functional explanations of the many unusual features of deep tracks. For example, the elongate front of deep Greenlandic tracks is almost identical to that made by living birds, such as turkeys, walking through deep mud. This verifies that the toes of early theropods converged as they were lifted from the substrate, a feature still retained by many birds today. Other features provide evidence of important differences. Birds leave no sole prints because they quickly lift their ankles after contacting the ground. In contrast, all deep tracks from early theropods have a substantial sole print, indicating that the ankle did not rise up until much later in the stride cycle. Finally, the impression left by the first toe (our big toe) points backward in Greenlandic tracks, whereas the toe itself points forward in all fossil skeletons of this age. A reversed first toe is characteristic of perching birds and not expected in such ancient theropods. However, three-dimensional computer simulations using a particle system reveal that a forward pointing first toe can create a backward pointing slash in the surface as it plunges down and forward into the mud.

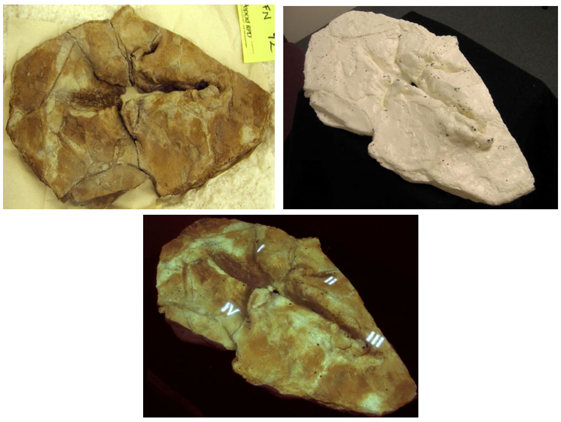


Figure 5.2: *Projector-based augmentation of a theropod track: The original track, preserved in Greenland's Triassic Fleming Fjord formation (upper-left). Hand-made cast of the shallow track (upper-right). Texture of the original track, with additional annotations that indicate the imprints of the toes, projected directly onto the cast (bottom).* [13]

In many cases original fossils are not available to museums or even the Paleontologists. In most cases fossils are shared between various institutions and researchers. Also due to local political circumstances or the possibility that the fossil would be damaged or destroyed by removing it, the original fossil may not be available.

In these situations, manmade casts are often used to substitute fossils. A cast can easily express the fossil's shape but not it's original texture and detail information, such as skin impressions. However this information can be captured with photographs.

Instead of painting the cast with a texture, video-projectors are applied to seamlessly map existing photographs and rendered images directly onto the cast's surface [13]. Such a projector-based augmentation in combination with physical replica allows us to recreate artifacts in laboratories and museums that are hundreds or thousands of miles away from the original (cf. figure 5.2).

A projector-based augmentation is also interactive. The lighting conditions and the surfaces' color properties can be changed dynamically. Additional information, such as annotations and highlights, can be displayed directly on the cast as part of an interactive or linear story-telling installation. In contrast to a completely virtual model (e.g., a textured 3D scan), a physical cast provides haptic and tactile feedback and a natural auto-stereoscopic three-dimensional perception.

5.1.2. AR and digital storytelling: from the scientist to the museum visitor

Interactive digital storytelling techniques are recently being applied in combination with new media forms, such as virtual reality (VR) and augmented reality (AR).

Thereby the technological progress that is being made within these areas allows shifting interactive digital storytelling more and more into the third dimension and into the physical world.

One of the main advantages of this transition is the possibility to communicate information more effectively with digital means by telling stories that can be experienced directly within a real environment or in combination with physical objects. The user experience is thus transformed from relating different pieces of information to one another to 'living through' the narrative.

The perceptual quality and the unique aura of a real environment (e.g., a historical site) or object (e.g., an ancient artifact) cannot be simulated by today's technology. Thus it is not possible to substitute them with virtual or electronic copies without them losing their flair of originality.

This circumstance can be a crucial reason for using augmented reality as a technological basis for interactive digital storytelling.

In [15] we show how the Virtual Showcase can be used as a new digital storytelling platform. We have identified five major components: content generation, authoring, presentation, interaction, and content management. We also describe examples of their applications.

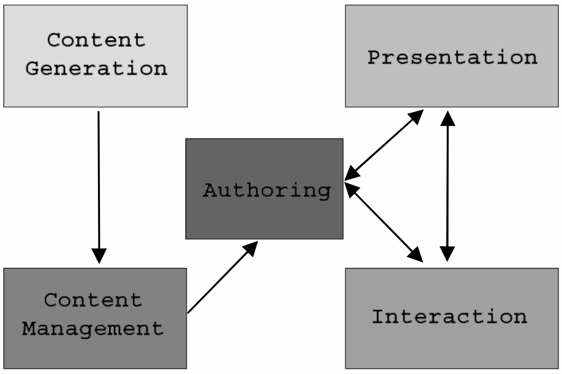


Figure 5.3: *Storytelling components and their dependencies* [15].

To describe details about the specific implementations of these components and how they interact is out of the focus of this tutorial. More information can be found in [15].

Conventional content types (such as 3D models, animations, and audio, etc.) can be generated, authored and presented with spatial AR displays that are the same as the content types that are essential for traditional or VR digital storytelling. More interesting, however, are the unconventional content types that are specific to augmented reality, or even to spatial augmented reality. For example, static and dynamic illumination of the real objects is as such an unconventional content that can be created, authored, managed, interacted with and presented within a digital story.

We focused on the soft-tissue reconstruction work of Witmer and his colleagues to demonstrate and evaluate the Virtual Showcase's storytelling capabilities. The goal was to explain the scientific findings of the palaeontologists to a non-scientific audience in an exciting and interesting way. Several layers of soft-tissue where overlaid over the physical skull and animated to illustrate their functions. Projector-based illumination was applied to create consistent occlusion effects between muscles and bones. Text labels and audio narrations (a verbal explanation by the palaeontologists) were presented in sync to the animations, and two users were supported simultaneously.

To date, more than five-thousand users have seen this demonstration during several public events (Siggraph 2002, Learntech 2003, CeBit 2003, Cybernarium Nights 2003). As part of a user study, approximately 25% returned valid questionnaires that we handed out to investigate the acceptance and visual quality of our approach. The questions and the average answers are presented below.

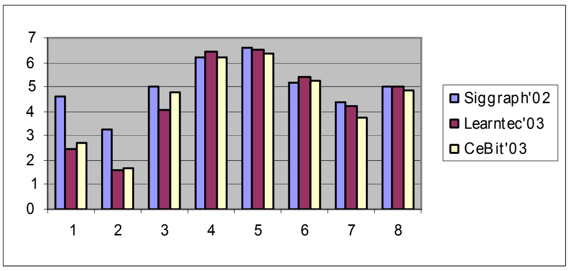


Figure 5.4: *User feedback on previous experience and acceptance* [15].

1. Do you have any previous experience with virtual reality? (1= none, 7= much)
2. Do you have any previous experience with augmented reality? (1= none, 7= much)
3. Do you have any previous experience with computer games? (1= none, 7= much)
4. Would you try out the same or a similar technology again? (1= not at all, 7= yes, very much so)
5. Do you think such technology is suitable for Museum exhibits? (1= not at all, 7= yes, very much so)
6. Did the virtual representation and the supporting technology deteriorate in any way your experience with the real object? (1= yes, very much so, 7= not at all)
7. Would you pay a higher entrance fee in order to see Virtual Showcase technology in a museum? (1= not at all, 7= definitely, if reasonable)
8. Would you prefer to go to a Virtual Showcase display rather than a traditional artifact exhibit of the same object in a museum? (1= not at all, 7= definitely)

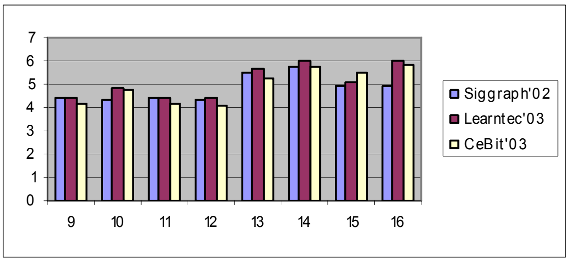


Figure 5.5: *User feedback on visual impression* [15].

9. How would you rate the comfort of the 3D glasses? (1= bad, 7= very good)
10. Did you have the impression that the virtual objects belonged to the real object (dinosaur skull), or did

they seem separate from it? (1= separate from the real object, 7= belonged to the real object)

11. Was watching the virtual objects just as natural as watching the real world? (1= completely unnatural, 7= completely natural)
12. Did you have the impression that you could have touched and grasped the virtual objects? (1= not at all, 7= absolutely)
13. Did the virtual objects appear to be (visualized) on a screen, or did you have the impression that they were located in space? (1= on screen, 7= in space)
14. Did you have the impression of seeing the virtual objects as merely flat images or as three-dimensional objects? (1= only as image, 7= as three-dimensional object)
15. Did you pay attention at all to the difference between real and virtual objects? (1= not at all, 7= yes, very much so)
16. Did you have to make an effort to recognize the virtual objects as being three-dimensional? (1= yes, very much so, 7= not at all)

The particular example that was described above illustrates how the Virtual Showcase was used successfully as a digital storytelling platform to present scientific findings within the palaeontology domain to a novice audience. Note that different versions of the Virtual Showcase have been, or will be set up in a variety of museums throughout Europe. Archaeological pieces have been exhibited in the Museo de San Telmo de San Sebastian, Spain from March 15th 2003 through June 15th 2003. Further Virtual Showcase exhibitions with archaeological, technical and bio-chemical contents will follow in the first half 2004. They will be hosted by the Deutsche Museum Bonn (Germany), the Technische Museum Wien (Austria), and the D. Diogo de Sousa museum, in Braga (Portugal).

5.2. Vehicle simulation, computer aided engineering and design

Let us consider the problem of creating the perception of motion without corresponding physical displacement in space. This perception of motion is known as *apparent motion*. Here we describe it in the context of simulation of motion for a car.

The motion analysis by humans is divided into three levels of processing: retinal motion sensing, 2D integration and 3D interpretation. The need to sense retinal motion, and analyze it as quickly as possible, places great demands on the visual system. A great deal of perceptual and physiological research has been conducted to discover the properties of these mechanisms. With several (retinal and other) cues, some potentially conflicting, the visual system attempts to integrate into a meaningful 'best' solution, which may be actually incorrect [70]. The resulting illusions provide us an opportunity.

Using 3D computer graphics, we exploit the induced motion effect and the 3D interpretation error for some very constrained cases. For induced motion, we segment a continuous static physical scene into sub-parts illuminated with images that have consistent rate of relative displacement. We also create temporary and invalid 3D interpretation using shadows, lighting and texture movements. We describe a system to show some limited effects on a static car model and present techniques that can be used in similar setups.

The scene is made up of a toy-car model, with a simple horizontal surface and a vertical surface. The horizontal surface coarsely represents the ground surface, either a road or rough terrain. The vertical surface is the backdrop which represents everything else including bushes, trees and sky. The background can be made arbitrarily complex (but must have constant profile along any plane perpendicular to the direction of apparent motion) to improve the visual fidelity.

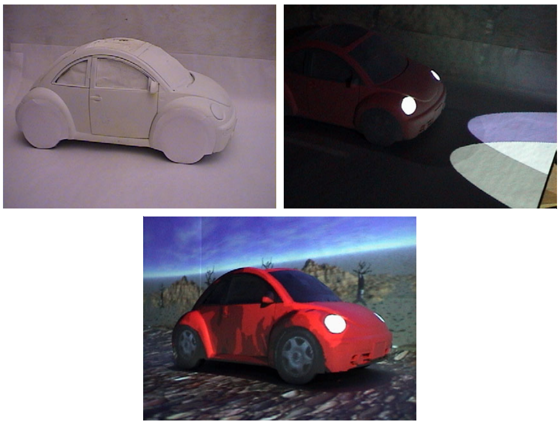


Figure 5.6: *The diorama of a car and approximate background. A cartoon-like environment created by image projection in which the car appears to move (bottom). A night-time simulation using virtual headlights (upper-right).* [100]

5.2.1. Motion effects

The car is simulated as driving along a road, on a rough surface, or in various other environments. To create apparent motion, we illuminate wheels with images of rotating wheels. The images of the background (made up of the backdrop and ground) move in a direction opposite to the intended car movement. In the simplest case, as seen in the video, the car appears to move forward i.e. left to right. To create this effect, the wheels rotate clockwise and the background moves right to left (Figure 5.7 (upper left)). The crucial task for a believable movement is maintaining consistency between the angular wheel movement and corresponding translation of the ground (and the backdrop). For any duration,

$$\int \text{wheel perimeter arclength} = \left|\, \text{displacement of the background} \,\right|$$

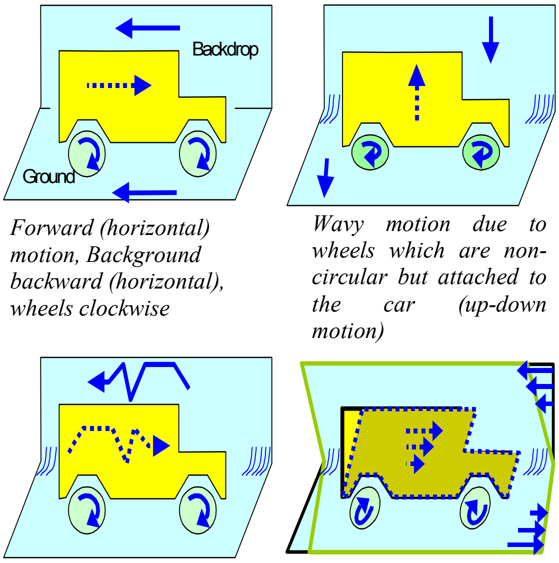
This ensures that the wheels are not sliding while the car is in motion. A small amount of motion blur is added to the wheels. Along the direction of the motion, the background geometry with the associated texture maps is infinitely long, and is implemented using a simple sink and a source.

We experimented with many types of cartoon or non-realistic motions. Two important factors that add to the effect are sound and removal of (physical) sharp edges in the background.

Wobbly motion

Slow moving cartoon cars usually have a wavy or bumpy movement resulting in a small periodic or random displacement along the vertical direction. This sometimes is emphasized by non-circular wheels. In our case, the car is static. Hence, to create the desired apparent vertical shift while the car is in motion, we instead translate the background in the opposite (vertical) direction (Figure 5.7 (upper right)). During rendering, the 3D model of the backdrop as well as the ground is translated vertically. The amount of translation, in the case of non-circular wheels, is determined by distance between the point of contact of the wheels from wheel axis (Figure 5.7 (upper right)). The distance traveled is again consistent with integration of the arc length of wheel perimeter. Both types of motions can be seen in the video available on our website. (The video does not do justice and it is difficult to feel the transitions, the effect has to be experienced in person with all the three dimensional cues. Please see the subsection on user reactions.)

It is important to note that to create apparent motion, sometimes, the underlying physical geometry does not match up with the virtual model. For example, with vertical displacement, the boundary between vertical and horizontal surface (shown as sharp edge in Figure 5.7 (upper left)) does not correspond to the projection of the boundary between the backdrop and ground. Hence, first we need to eliminate any sharp edges that the user may be able to use as a frame of reference. Second, we need to represent the surface with the approximate average of corresponding variation in the geometry. The solution is to the smooth the edge using a paper with small curvature (see Figure 5.7 (upper right)).



Forward (horizontal) motion, Background backward (horizontal), wheels clockwise

Wavy motion due to wheels which are non-circular but attached to the car (up-down motion)

Bumpy movement on a rough surface

Shear during acceleration

Figure 5.7: *Types of apparent motion for a car. Rendered motion is shown with blue arrow. The resultant apparent motion is shown with dashed blue arrow.* [100]

Lights and shadows

Shadows provide a very important cue in apparent motion. For example, in a two-dimensional image sequence (i.e. without any other depth cues), a ball moving diagonally upwards across the screen can be made to appear as if it is gradually moving away from the viewer by rendering a shadow that appears to stay with and under the ball. The cue overrides contradictory information regarding the unchanging size of the ball. The strength of the effect does depend on the properties of the shadow region. For example, shadows with a penumbra, and which fall below the object, work best.

According to [55] the existence of the illusion shows that our perception of the spatial layout of scenes relies on the assumption that the light source responsible for the shadow is stationary, so that any motion of the shadow is attributed to the object casting it.

We enhance the wobbly motion effect by rendering virtual shadows from directional light source (sun) or local lights (street lights). The change in shadow position creates the illusion that the change is a result of changing vertical distance between the car and the ground. We noticed that, it is not necessary to use the same light position to calculate shading and shadows ! Further, perceptually, the movement of virtual shadows (Figure 5.8) is not affected by the fixed real shadows of the physical car on the background.

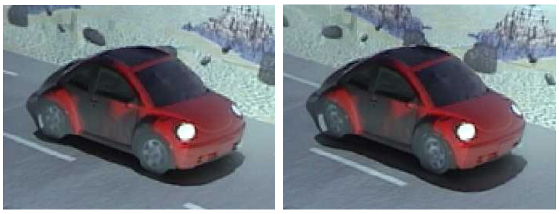


Figure 5.8: *Vertical displacement by shifting the background and shadows. These two pictures are taken from the same camera position, so the car in both images is at the same location. Note the change in position of the virtual shadows and the parallax for the edge between ground and backdrop.* [100]

For night time simulation, the only cues are headlight beams and shading due to the street lights. The spread of the parabolic projection of the headlights and change in overlap between the two beams indicates the vertical shift. The color of the two beams is intentionally chosen slightly different, so that the lateral shift as the car moves up and down is clearly seen. We also exploit spot lights from the street lights (Figure 5.9).

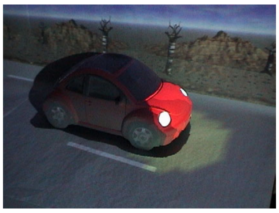


Figure 5.9: *Shading due to street lights as spot lights* [100].

Acceleration-dependent modifications

Cartoon animators emphasize acceleration, such as a sudden start or a screeching halt, by geometric deformation of the object. For example, a car appears to 'work hard' to move forward while starting when the top of the car is sheared in the direction of the acceleration. Similarly a hard brake and stop is indicated by shearing the top backwards. Since we cannot shear the physical model, we enhance this effect using two tricks.

First, we implement a shear in the background that is opposite of the shear expected in the car (Figure 5.7(iv)). The shear is along one dimension, along the vertical axis. Hence, for example, during a sudden start, the background shears backward and the shear at a point is proportional to the vertical distance from the center of the car. The points above the vertical center of the car translate backwards while points below translate forward (Figure 5.10). Without a loss of generality, lets assume that the car center is at the origin, the vertical direction is parallel to the z-axis, and the forward direction is parallel to the x-axis, Then the shear at a given frame is achieved using a simple transformation matrix [1, 0, -*a*, 0; 0, 1, 0, 0; 0, 0, 1, 0; 0, 0, 0, 1]. Where *a* is proportional to the acceleration at that frame.

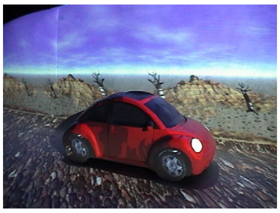


Figure 5.10: *Shearing of the background during acceleration* [100].

Since the acceleration is positive during starting, negative during braking, and zero during constant speed and velocity, the same shear equation can be used throughout the animation.

For the second trick, we observe that rapid acceleration also means relative slide between the ground and the wheels. Hence, for example, a sudden brake results in halt in rotation of the wheels, but the background continues to move (Please see the video available on our website).

Tracked illumination

We synchronize and update the rendering of the objects so that the surface textures appear glued to the objects even as they move. In this case, we rotate the car (along with the background) on a turntable. Keeping the virtual light sources fixed, we see corresponding shading changes.

Issues

The type of effects possible are obviously limited. For example, we played with motion blur and motion lines typically seen in cartoon animations, but without success. A simple problem is lack of corresponding physical surface to show motion lines behind the car. We also could not bring in effects such as fog, smoke or fumes. Displaying fog simply washes out the image on the model. During the night time simulation, the backdrop, which is supposed to be at a large depth and hence dark, gets illuminated by the secondary scattering of headlight beams.

5.2.2. User reactions

We do not have systematic user experience data but the system has been seen by several thousand people. The working system has been demonstrated at shows such as INFOCOMM (USA) and CEATEC (Japan), to visitors of the lab, at an art show, during UIST'01 conference, and to dozens of Disney researchers in Orlando, Florida. Most users get excited at the following stages: (i) when the car starts, (ii) when the car motion switches to bumpy motion, (iii) when the car is rotated on the turn table while maintaining its augmented surface appearance. As mentioned earlier, the effect is most noticeable during any changes in apparent motion and it lasts for a short time. The human visual motion detection system quickly becomes balanced again. Hence, it is important to constantly change the motion parameters. The synchronized car engine (or brake) sound seems to have a huge impact, it generates anticipation before the car starts and reinforces the notion of a car movement (as opposed to movement of the background). We implemented a photo-realistic version as well as the cartoon-like version (Figure 5.12). The cartoon-like version appeals more when the car is in motion. Viewers, not familiar with the technique, are unimpressed to see the static cartoon version. This is probably because we are all used to looking at much higher quality cartoon-like setups. But, the interest increased when we interactively changed the surface appearance of the car or the background. Most viewers, even after explaining that we are projecting on white surfaces, are very surprised when the projector light is blocked to show the underlying dull setup of simple static models. Comparing figure 5.11 and figure 5.12, it is surprising to see the amount fine detail possible with purely projected images.

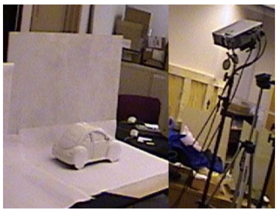


Figure 5.11: *Setup with projector and simple diorama* [100].

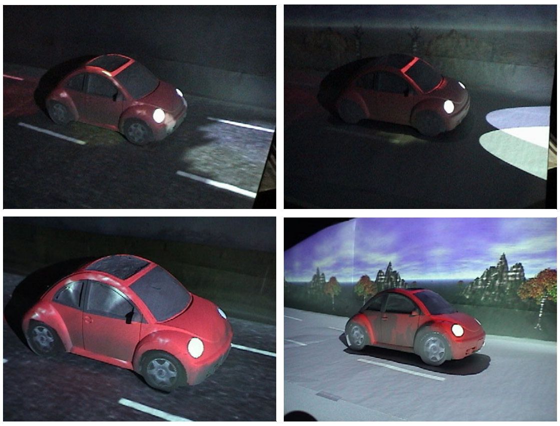


Figure 5.12: *Photo-realistic diorama (left column) vs. cartoon diorama (right column) appearance based on shading and motion.* [100]

So far, some viewers have complained of feeling dizzy (a well known side effect of VR/VE worlds). Children show a tendency to run and catch the moving car. Advanced technical viewers after careful observation sometimes notice the lack of complete illumination (for example, shadow of the car on the backdrop is not filled in by multiple projectors).

6. Summary and enabling technologies

In this tutorial we discussed application specific alternative augmented reality approaches that focus on overcoming some of the limitations linked to conventional AR displays.

We gave an extensive overview over different display techniques that may enable readers to identify parallels between virtual reality and augmented reality display technology, and stimulate them to think about alternative display approaches for AR. A classification and examples of goggle-bound stereoscopic, auto-stereoscopic, head-attached, handled, object oriented, spatial, optical see-through and projection-based displays was presented. Although auto-stereoscopic displays are not common for augmented reality today, we believe that they will play a major role in the future.

Augmented reality techniques using non-trivial projection screens were described. One of our emphasises was on projector-based augmented reality. In this context, we discussed two scenarios, a concept of spatial augmentation and then a more specific method of changing surface appearance of real objects.

Our other focus was on spatial optical see-through displays: We described two examples: The Extended Virtual Table as a single user setup, and the Virtual Showcase that supports multiple users. These two display concepts served as references for describing rendering and illumination techniques. Interactive rendering techniques that can be used to drive spatial optical see-through AR displays which consist of a combination of single or multiple beam-splitters (planar or curved) and screens (projection displays or monitors) were presented in detail. Finally, the projector-based illumination concept was described, and examples were outlined how it can be used to create consistent illumination and occlusion effects.

Two current areas of application of projector-based augmented reality and spatial optical see-through displays have been discussed: scientific visualization and digital storytelling within a museum context; and vehicle simulation, and computer aided engineering and design. Concrete setups and demonstrations were presented that have been displayed to a large audience during several public events. User feedback and reactions are presented and discussed.

We want to annotate that upcoming and new technology will not only open new possibilities for projection-based or spatial augmented reality, but also for other display concepts, such as hand-held and head-attached displays.

Projectors of the near future, for instance, will be compact, portable, and with the built-in awareness which will enable them to automatically create satisfactory displays on many of the surfaces in the everyday environment. Alongside the advantages, there are limitations, but we anticipate projectors being complementary to other modes of display for everyday personal use in the future, and to have new application areas for which they are especially suited. LEDs are replacing lamps and reflective instead of transmissive displays (DLPs, LCOS) are becoming popular. Both lead to improved efficiency requiring less power and less cooling. Several efforts are already in the making and are very promising. For example Symbol Technologies [117] has demonstrated a small laser projector (two tiny steering mirrors for vertical and horizontal deflection) and has even built a handheld 3D scanner based on such a projector. Siemens has built a 'mini-beamer' attachment for mobile-phones [107]. Cam3D has built a 'Wedge' display where a projector can be converted into a 'flat panel' display by projecting images at the bottom of a wedge shaped glass [123]. A future mobile projector may double up as 'flat panel' when there is no appropriate surface to illuminate, or ambient light is problematic. Super bright, sharp infinite focus laser projectors are also becoming widespread [51] which may allow shape-adaptive projection without focus and ambient lighting problems. In addition suitable input devices are also appearing; e.g., Canesta [22] has built a projected laser pattern on which one can type. The finger movement is detected by IR sensing. Finally novel lamp designs, especially those based on LEDs or lasers are creating smaller, lighter, efficient and long-life solutions.

Organic Light Emitting Diodes (OLEDs) [47], 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.

In contrast to normal LEDs, OLEDs are made from plastic compounds instead from semi-conducting elements, such as silicon or gallium, etc. Like LEDs, OLEDs glow when voltage is applied. Two main classes exist today: small molecule OLEDs and polymer OLEDs.

While small molecule OLEDs are built up by depositing molecules of the compound onto the display itself under very low pressure, polymer OLEDs have the active molecules suspended in a liquid-like pigment in paint. It can be printed onto displays using ink jets, screen printing or any of the various contact techniques used for ordinary inks. Small molecule OLED displays are limited in size, but they may be suitable for head-mounted displays. Polymer OLEDs can be used to build large scale displays – such as 500 inch displays or larger. Resolution approaching 300 dpi is also possible, approaching the quality of ink on paper. They may become more interesting for spatial AR approaches.

The general advantages of OLEDs are:

- Because OLED displays are fluorescent and don't require backlights they will need far less power than LCD screens (currently LCDs require three times as much power than OLEDs);
- OLED layers can be made much thinner than LCD layers (about a thousand times thinner than a human hair);

- In contrast to LCDs, OLEDs don't use polarized light filters. The displayed images on OLEDs can be viewed from a much wider angle;
- OLEDs have a much wider working temperature range than LCDs;
- Can be "printed" onto large-scale and flexible display surfaces of any shape and (almost) any material.

Today's reality, however, is that the OLED compounds degrade over time (especially when they get in contact with oxygen or water) – limiting the maximum lifetime of a display. Different colors degrade at different rates – making the color balance change. These problems may be solved in future.

A variation of OLEDs are Light Emitting Polymers (LEPs) [21] that provide the opportunity for the fabrication of large, flexible, full-color, fast emissive displays with a high resolution, a wide viewing angle and a high durability.

In LEP technology, a thin film of light-emitting polymer is applied onto a glass or plastic substrate coated with a transparent electrode. A metal electrode is evaporated on top of the polymer. The polymer emits light when the electric field between the two electrodes is activated.

The response time of LEPs is ultra-fast (sub-microsecond) and is unaffected by temperature. Consequently they may support high enough frame rates for active stereoscopic rendering. Light emission occurs at low voltage, and (unlike LCD or plasma displays) it can be fabricated on a single sheet of glass. Also, because it can be made of flexible plastic substrates, it is not only extremely difficult to break, but can also be molded into different shapes and contours.

The advantages of LEPs can be summarized as follows:

- Since a low voltage is required LEPs need little power;
- The response time of LEPs is very high – potentially allowing active stereoscopic rendering;
- Can be fabricated on transparent glass or plastic surfaces of any shape;
- LEPs provide a high contrast (currently between 3-5 times higher than LCDs).

Transparent LEPs may present other future alternatives for spatial AR configurations.

Electronic paper (or E-Ink) [131] also has potential as an AR display technology of the future. Here an electric charge moves magnetic colored capsules within the "paper" either toward or away from the surface in order to form an image. The capsules retain their positions until another charge is applied. The ink simply resides on the display while an image is not changing; therefore, it consumes no power. Philips and other companies in the field are working on color as well as bendable applications.

The main advantage of electronic paper is that it does not require power at all, as long as the displayed image does not change.

The current generation of electronic paper, however, is a black-on-white technology where the ink is pumped onto a white background, giving maximum readability.

Solid-state volumetric displays [65] have already been mentioned in section 2.1.2. They generate visible photons (i.e. light) within a transparent host material by exciting optically active ions with special energy sources. Ions at a known three-dimensional position within the host materials can be excited by crossing energy beams, such as infrared lasers, ultraviolet sources of radiation, or electron beams.

Examples of suitable host materials are various gases, crystals and electro-active polymers. The host material must provide several properties:

- In its initial state it must be transparent;
- It must emit visible light in its excited state;
- Its inner structure must be homogenous;
- It must have a refraction index similar to air to avoid distortion.

If it would be possible to use air as the host material, then this approach would represent the holy grail – not only for spatial augmented reality displays, but for 3D display technology in general. Unfortunately this is not yet feasible. In addition, conceptual problems, such as the "ghost voxel problem" (when energy beams that are used to create voxels intersect with other beams and create unwanted voxels) have to be solved.

We believe that the application of parallax displays (see section 2.1.3) will be the next logical and feasible step for spatial AR solution towards autostereoscopy. For instance, we are currently working on a version of the Virtual Showcase that uses lenticular sheet displays and eye-tracking, instead of CRT screens, shutter glasses and head-tracking. This will allow detaching the display technology from the users completely.

In the future, a combination of computer graphics and holography can also be expected. While holographic images of larger ensembles hardly seem feasible for technical reasons, any three-dimensionally programmed computer image can, in principle, be transformed into a hologram basically by way of Fourier transformations. These, however, presuppose large computational, network and storage capacities. For instance, a pre-rendered or recorded movie-length holographic video could require a petabyte (1 million gigabytes) of storage. Holographic storage itself, which uses lasers to record data in the three dimensions of a clear crystalline medium may be a future solution to the problem that non-static holograms require a massive amount of data. While computer generated static images can already be transformed into large digital holograms using holographic printers [59], interactive or real-time electronic holography [69] with an acceptable quality (size, resolution and colors) would still require the invention of more advanced light modulators, faster computers with a higher bandwidth, and better compression techniques.

In the short run, especially high-resolution bright and flexible projection devices, high-performance and cost-efficient rendering hardware, reliable, precise and wireless tracking technology, and advanced interaction techniques and devices will pave the way for forthcoming alternative AR configurations. In the long run, new display concepts, such as auto-stereoscopy and holography will replace goggle-bound stereoscopic displays (at least for non-mobile applications). 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 augmented reality of the future.

Acknowledgements

We would like to acknowledge and thank all contributors of this tutorial: M. Agrawala, R. T. Azuma, E. Foxlin, G. Goebbels, H. Hua, M. Inami, N. Kawakami, Y. Kitamura, K. Kiyokawa, K. Low, J. Newman, T. Ogi, C. Pinhanez, A. Preston, B. Schwald, I. Siio, G. Stetten and G. Wetzstein.

The Virtual Showcase project is supported by the European Union (IST-2001-28610).

References

1. 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.

2. 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 Workshop (IPT'98), 1998.

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

4. 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.

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

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

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

8. 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.

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

10. 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.

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

12. 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), pp. 186-195, 2002.

13. 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.
14. Bimber, O. Interactive rendering for Projection-Based Augmented Reality Displays. Ph.D. Dissertation, University of Technology Darmstadt, http://elib.tu-darmstadt.de/diss/000270/, 2002.
15. Bimber, O., Encarnação, L.M., and Schmalstieg, D. The Virtual Showcase as a new Platform for Augmented Reality Digital Storytelling. To appear in proceedings of IPT/EGVE 2003 workshop, 2003.
16. Bimber, O., Fröhlich, B., Schmalstieg, D., and Encarnação, L.M. Real-Time View-Dependent Image Warping to correct Non-Linear Distortion for Curved Virtual Showcase Displays. To appear in Computers and Graphics - The international Journal of Systems and Applications in Computer Graphics, vol. 27, no. 4, 2003.
17. Bimber, O., Grundhöfer, A., Wetzstein, G., and Knödel, S. Consistent Illumination within Optical See-Through Augmented Environments. Submitted to International Symposium on Mixed and Augmented Reality (ISMAR'03), 2003.
18. Blundell, B. and Schwarz, A. Volumetric Three-dimmensional Display Systems. New York: Wiley-Interscience, 2000.
19. Bandyopadhyay, D., Raskar, R., State, A., Fuchs, H., Dynamic Spatially Augmented 3D Painting. UNC Chapel Hill Tech Report TR01-006, 2001.
20. 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.
21. Burroughes, J.H. Light Emitting Polymers. Nature, pp. 347, 1990.
22. Canesta. Miniature Laser Projector Keyboard. Retrieved from the World Wide Web: http://www.canesta.com. 2002.
23. Chen, S. E. and Williams, L. View Interpolation from Image Synthesis. SIGGRAPH '93, pp. 279-288, July 1993.
24. Chinnock, C. Holographic 3-D images float in free space. Laser Focus World, vol. 31, no. 6, pp. 22-24, 1995.
25. Cruz-Neira, C., Sandin, D., and DeFanti, T. Surround-Screen Projection-Based Virtual Reality: The Design and Implementation of the CAVE. Computer Graphics (proceedings of SIGGRAPH'93), pp. 135-142, 1993.
26. Debevec, P., Taylor, C. J., and Malik, J. Modeling and Rendering Architecture from Photographs. SIGGRAPH '96, August 1996.
27. Debevec, P., Yu, Y., and Borshukov, G. Efficient View-Dependent Image-Based Rendering with Projective Texture-Mapping. Proc. of 9th Eurographics Workshop on Rendering, June 1998.
28. Dimensional Media Associates, Inc., Retrieved from the World Wide Web: http://www.3dmedia.com/, 2001.
29. Dorsey, J., Sillion, F. X., and Greenberg, D. Design and Simulation of Opera Lighting and Projection Effects. SIGGRAPH '91, August 1991.
30. 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.
31. Elings, V. B. and Landry, C. J. Optical display device. U.S. Patent, No. 3,647,284, 1972.
32. Fakespace Labs, Inc. BOOM 3C. Retrieved from the World Wide Web: http://www.fakespacelabs.com/products/boom3c.html, 2001.
33. Fakespace Systems, Inc. ImmersaDesk Series. Retrieved from the World Wide Web: http://www.fakespacesystems.com/products.html, 2001.
34. Faugeras, O. Three-Dimensional Computer Vision: A Geometric Viewpoint. MIT Press, Cambridge, Massachusetts, 1993.
35. Favalora, G.E. Multiplanar autostereoscopic imaging system. U.S. Patent, No.5,936,767, 1999.
36. 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.

37. 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.

38. Figl, M., Birkfellner, W., Ede, C., Hummel, J., Hanel, R. Watzinger, F., Wanschitz, F., Ewers, R., and Bergmann H. The Control Unit for a Head Mounted Operating Microscope used for Augmented Reality Visualization in Computer Aided Surgery. In proceedings of International Symposium on Mixed and Augmented Reality (ISMAR'02), pp. 69-76, 2002.

39. 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.

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

41. Gatesy, S.M., Middleton, K.M., Jenkins, F.A., Jr. and Shubin, N.H. Three-dimensional preservation of foot movement in Triassic theropod dinosaurs. Nature, vol. 399, pp. 141-144, 1999.

42. Gortler, S. J., Grzeszczuk, R., Szeliski, R., and Cohen, M. F. The Lumigraph. SIGGRAPH '96, August 1996.

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

44. Heckbert, P. and Hanrahan, P. Beam tracing polygonal objects. Compute Graphics (proceedings of SIGGRAPH'84), vol. 18, no. 3, pp. 119-127, 1984.

45. 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.

46. HowardModels.com, 7944 Central Avenue, Toledo, OH 43617. http://www.howardweb.com/model/.

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

48. Hua, H., Gao, C., Brown, L., Ahuja, N., and Rolland, J.P. Using a head-mounted projective display in interactive augmented environments. In Proceedings of IEEE and ACM International Symposium on Augmented Reality 2001, pp. 217-223, 2001.

49. The HyperMask project. http://www.csl.sony.co.jp/person/nielsen/HYPERMASK/.

50. Inami, M., Kawakami, N., Sekiguchi, D., Yanagida, Y., Maeda, T. and Tachi, S., Visuo-Haptic Display Using Head-Mounted Projector, Proceedings of IEEE Virtual Reality 2000, pp.233-240, 2000.

51. Jenoptik. Laser Projector. Retrieved from the World Wide Web: http://www.jenoptik.com, 2002.

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

53. Kajiya, J. T. The Rendering Equation. Computer Graphics 20(4) (1986), pp. 143-151.

54. Kawakami, N., Inami, M., Sekiguchi, D., Yanagida, Y., Maeda, T. and Tachi, S. Object-Oriented Displays: A New Type of Display Systems -From immersive display to Object-Oriented Displays-, IEEE-SMC '99 Abstracts, Tokyo, Japan, p. 493, October 1999.

55. Kersten D, Mamassian P, Knill D. Moving cast shadows induce apparent motion in depth. Perception, vol. 26, pp 171-192, 1997.

56. 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.

57. 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.

58. 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.

59. Klug, M.A., Display applications of large scale digital holography. In Proceedings of SPIE, vol. 4737, Holography: A Tribute to Yuri Denisyuk and Emmett Leith, 2002.

60. Knoll, W. and Hechinger, M. Architectural Models: Construction Techniques. McGraw-Hill Publishing Company, ISBN 0-07-071543-2.

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

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

63. Krueger, W., and Fröhlich, B. The responsive workbench. IEEE Computer Graphics and Applications, pp. 12-15, 1994.

64. 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.

65. Langhans, K., Guill, C., Rieper, E. Oltmann, K., and Bahr, D. Solid FELIX: A static 3D-Laser Display, In Proceedings of Symposium on Electronic Imagining: Science and Technology (IS&T/SPIE), vol. 5006, 2003.

66. Levoy, M. and Hanrahan, P. Light Field Rendering. SIGGRAPH '96, August 1996.

67. Liljegren, G. E. and Foster, E. L. Figure with Back Projected Image Using Fiber Optics. US Patent # 4,978.216, Walt Disney Company, USA, December 1990.

68. Low, K., Welch, G., Lastra, A., and Fuchs, H. Life-Sized Projector-Based Dioramas. Symposium on Virtual Reality Software and Technology, 2001.

69. 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.

70. Mather G, Verstraten F, and Anstis S The Motion Aftereffect: a Modern Perspective. Cambridge, Mass: MIT Press, 1998.

71. Majumder, A., He, Z., Towles, H., and Welch, G. Color Calibration of Projectors for Large Tiled Displays, IEEE Visualization 2000.

72. 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.

73. 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.

74. McKay, S., Mason, S., Mair, L. S., Waddell, P., and Fraser, M. Stereoscopic Display using a 1.2-M Diameter Stretchable Membrane Mirror. In proceedings of SPIE, vol. 3639, pp. 122-131, 1999.

75. McMillan, L. and Bishop, G. Plenoptic Modeling. SIGGRAPH '95, August 1995.

76. 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.

77. Milgram, P. and Kishino, F. A Taxonomy of Mixed Reality Visual Displays. IEICE Transactions on Information Systems E77-D, vol.12, pp. 1321-1329, 1994.

78. 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

79. Mizuno, G. Display device. U.S. Patent, No. 4,776,118, 1988.

80. Naemura, T., Nitta, T., Mimura, A., Harashima, H. Virtual Shadows – Enhanced Interaction in Mixed Reality Environments. In proceedings of IEEE Virtual Reality (IEEE VR'02), pp. 293-294, 2002.

81. Naimark, M. Displacements. An exhibit at the San Francisco Museum of Modern Art, San Francisco, CA (USA), 1984.

82. Newman, J., Ingram, D., and Hopper, A. Augmented Reality in a Wide Area Sentient Environment, In proceedings of IEEE/ACM International Symposium on Augmented Reality (ISMAR'01), , 2001, pp. 77-86.

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

84. 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.

85. O'Rourke, J. Art Gallery Theorems and Algorithms. Oxford University Press, New York, 1987.

86. Panoram Technology. http://www.panoramtech.com

87. 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.

88. 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.

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

90. Pinhanez, C. (2001). The everywhere displays projector: A device to create ubiquitous graphical interfaces. In Proc. of Ubiquitous Computing 2001 (Ubicomp'01), Atlanta, Georgia.

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

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

93. 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.

94. 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.

95. 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.

96. 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.

97. 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.

98. Raskar, R., Brown, M., Yang, R., Chen W., Welch, G., Towles, H., Seales, B., Fuchs, H.. Multi-Projector Displays Using Camera-Based Registration. IEEE Visualization 99, October 1999.

99. 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.

100. Raskar, R., Ziegler, R, Willwacher, T. Cartoon Dioramas in Motion. Int Symp on Non-photorealistic Animation and Rendering (NPAR), Annecy, France, 2002

101. Raskar, R. van Baar, J., Beardsly, P, Willwacher, T., Rao, S., and Forlines, C. iLamps: Geometrically Aware and Self-Configuring Projectors. To appear in Computer Graphics (proceedings of SIGGRAPH'03), 2003.

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

103. 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.

104. 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.

105. Schwald, B., Seibert, H., Weller, T. A Flexible Tracking Concept Applied to Medical Scenarios Using an AR Window. In proceedings of International Symposium on Mixed and Augmented Reality (ISMAR'02), pp. 261-262, 2002.

106. 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.

107. Siemens. Siemens Mini Beamer. Retrieved from the World Wide Web: http://w4.siemens.de/en2/html/press//newsdesk archive/2002/-foe02121b.html, Cited December 2002.

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

109. Siio, I., Virtual Glassboat: For looking under the Ground. Human-Computer Interaction International (HCII'01), 2001, pp. 683-687

110. Smith A, and Snowden R Visual Detection of Motion. London: Academic Press, 1994.

111. 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.

112. Stetten, G., Chib, V., Hildebrand, D., and 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.

113. Stuerzlinger, W. Imaging all Visible Surfaces. Graphics Interface '99, pp. 115-122, June 1999.

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

115. Sutherland, I.E. The Ultimate Display. In proceedings of IFIP'65, 506-508, 1965.

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

117. Symbol. Laser Projection Display. Retrieved from the World Wide Web: http://www.symbol.com/products/oem/lpd.html, December, 2002.

118. Szeliski, R. and Shum, H. Creating Full View Panoramic Mosaics and Environment Maps. SIGGRAPH '97, August 1997.

119. TAN Projectiontechnologies. Tanorama Cylinder. Retrieved from the World Wide Web: http://www.tan.de, 2001.

120. TAN Projectiontechnologies. Responsive Workbench. Retrieved from the World Wide Web: http://www.tan.de, 2001.

121. TAN Projectiontechnologies. Holobench. Retrieved from the World Wide Web: http://www.tan.de, 2001.

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

123. Travis, A., Payne, F., Zhong, J., and More, J. Flat panel display using projection within a wedge-shaped waveguide. Retrieved from the World Wide Web: http://ds.dial.pipex.com/cam3d/technology/technology01.html, 2002.

124. Underkoffler, J., Ullmer, B., and Ishii, H. Emancipated pixels: real-world graphics in the luminous room. SIGGRAPH '99, August 1999.

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

126. 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.

127. 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.

128. Wellner, B. Interacting with paper on the DigitalDesk. Communications of the ACM, 36(7):87–96, July 1993.

129. 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.

130. Witmer, L. M. The evolution of the antorbital cavity of archosaurs: a study in soft-tissue reconstruction in the fossil record with an analysis of the function of pneumaticity. Journal of Vertebrate Paleontology, vol. 17, supplement to no. 3, pp. 1-73, 1997.

131. Xerox PARC, Electronic Reusable Paper. Retrieved from the World Wide Web: http://www2.parc.com/dhl/projects/gyricon/ , 2003.

Appendix 1: Construction of a Monitor-Based Virtual Showcase

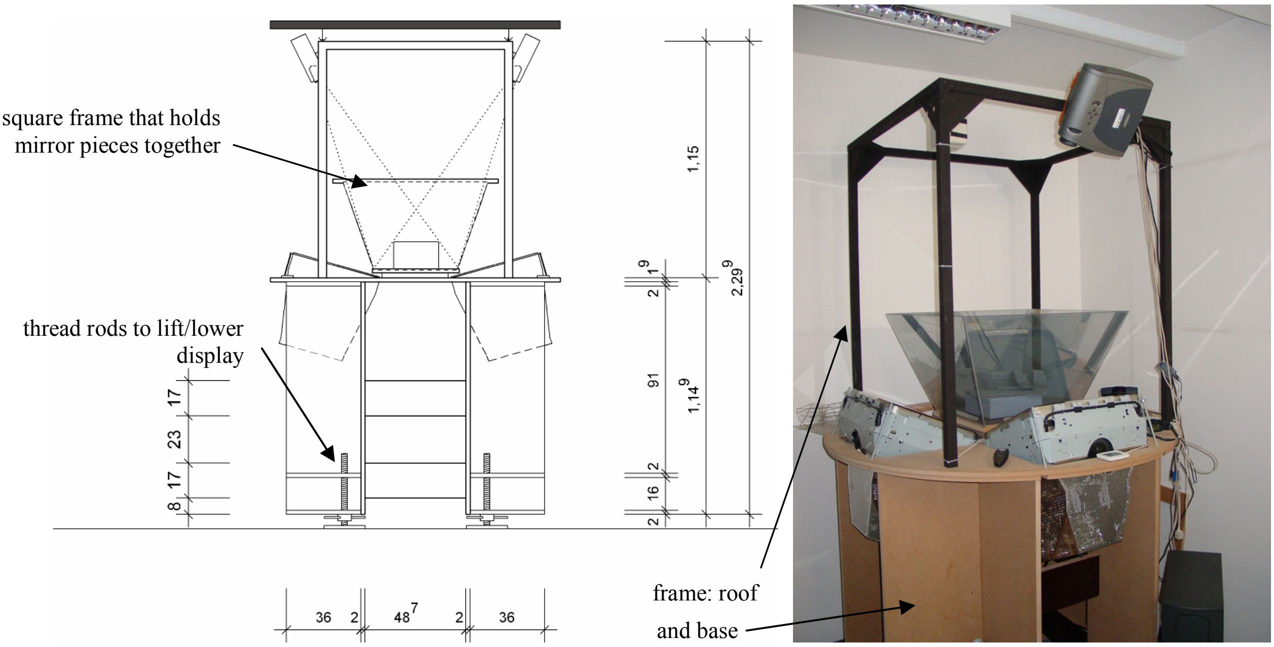
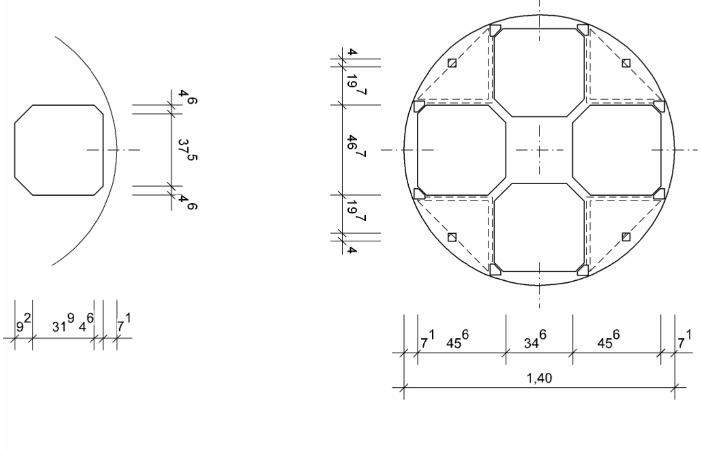

square frame that holds mirror pieces together
thread rods to lift/lower display
17
23
17
8
1,15
9
1
2
2,29
91
1,14
2
16
2
36
2
48
7
2
36
frame: roof
and base

Fig A1.1: Side *view – showing frame that holds monitors, mirror optics, and roof, roof holds video projectors.*



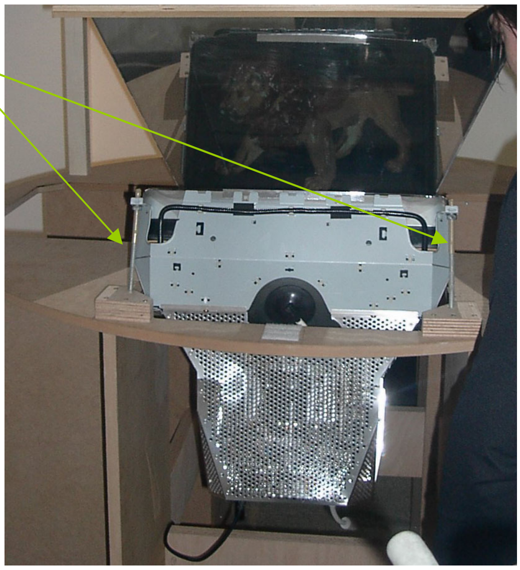


Fig A1.2: *Top-down view – showing base that holds monitors.*

Fig. A1.3: *Side view – showing how monitors are assembled into frame.*

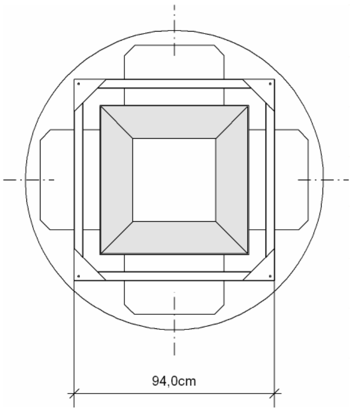


Fig A1.4: *Top-down view - shownig base (with mirror optics and roof).*

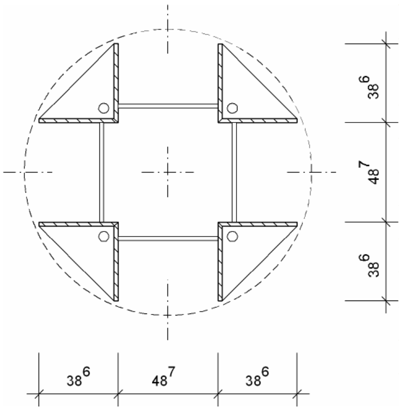


Fig A1.5: *Bottom-up view - showing base.*

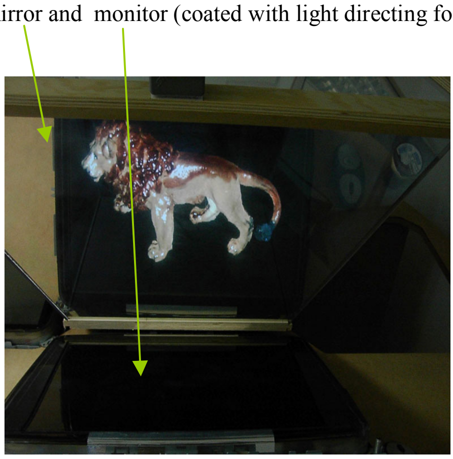

mirror and monitor (coated with light directing foil)

Fig A1.6: *Monitor coated with light-directing film.*

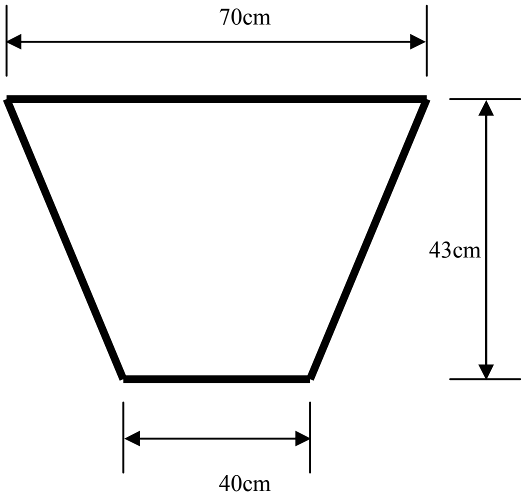
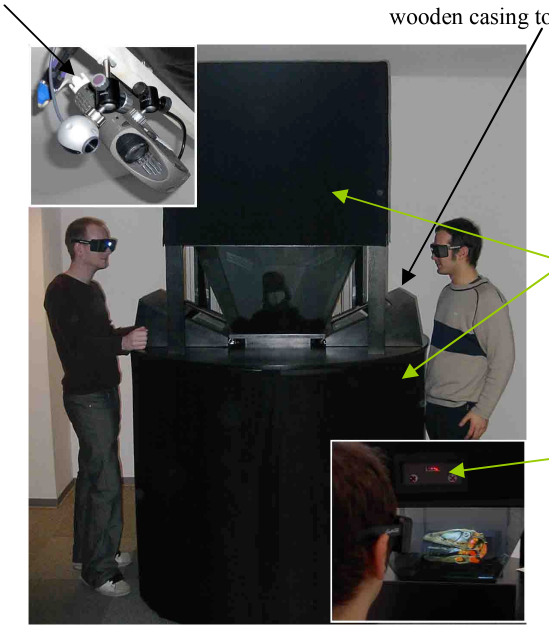


Fig A1.7: *Dimensions of single mirror (3mm-5mm thick).*


video projector and (optional) video camera mounted into rood

wooden casing to hide monitors

draped frame and roof to hide technology

tracking system mounted into roof

Fig A1.8: Final *(2-user) version with casing and tracking system.*

Currently used Hardware Components:

Quantity	Component	Currently used Type	Features	Comment
1-4	monitor	Sony GDM-F520	21", 1600x1200@100Hz	frame has to be removed, equivalent CRT monitors can be used (but frame might have to be adapted to new shape)
1-4	shutter glasses	StereoGraphics CrystalEyes	Wireless/IR	equivalent glasses can be used
1-4	IR emitter	StereoGraphics	requires BNC sync and cable with 3-pin connector for PC	equivalent emitters can be used
1-4	video projectors	Anders+Kern Astro-Beam x10	DLP, at least 1024x768,	equivalent beamer can be used
0-4	video cameras	Logitech QuickCam Pro3000/4000	USB or FireWire, at least 640x480 (true)	equivalent cameras can be used, optional
1-4	tracking system	Origin Dynasight	IR position tracking system	equivalent tracking system can be used
1-3	PCs	self-assembled	Minimum: pentium 4, >=1.8GhZ, GeForce4Quadro, 2 USB, 1 Serial, >=512MB Ram	1 PC is used to drive 2 displays – either monitor or video beamer
4	mirror beam splitters	self-made: Plexiglas of glass coated with half-silvered mirror foil	half-silvered mirror foil: 3M Scotchtint	Glass is better than Plexiglas (more stable), plates between 3-5mm, 3M Scotchtint foil is normally sun protection foil (not optimal), optimal → 50%/50% (reflection/transmission), good alternative is spy glass
1-4	light directing foil for monitors	3M light control film (available at Edmund Optics)	0 deg. louver angle (for tilted monitors)	louver angle depends on monitor angle, our current monitor angle is 30 deg (measured horizontally)

Appendix 2: Calibration Code for a Projector/Camera

Input: set of corresponding 3D points points.{X,Y, Z} and 2D pixels pixels.{u,v}
Output of step 3: Perspective 3 by 4 projection matrix, ProjMV
Output of step 3: OpenGL 4x4 matrices, ProjMat, ModelViewMat

```
void calculateMatrices()
{
 // Step 1: Fill Matrix that covers the constraining equations
 Matrix lhs(2*nPointsFound, 12); // lhs=LeftHandSide
 for (int i = 0; i < nPointsFound; i++) {
      // odd rows
      lhs(2*i+1, 1)= points[i].X;
      lhs(2*i+1, 2)= points[i].Y;
      lhs(2*i+1, 3)= points[i].Z;
      lhs(2*i+1, 4)=1;
      lhs(2*i+1, 5)=0; lhs(2*i+1, 6)=0;
      lhs(2*i+1, 7)=0; lhs(2*i+1, 8)=0;
      lhs(2*i+1, 9)= -pixels[i].u * points[i].X;
      lhs(2*i+1, 10)= -pixels[i].u * points[i].Y;
      lhs(2*i+1, 11)= -pixels[i].u * points[i].Z;
      lhs(2*i+1, 12)= -pixels[i].u;
      // even rows
      lhs(2*i+2, 1)=0; lhs(2*i+2, 2)=0;
      lhs(2*i+2, 3)=0; lhs(2*i+2, 4)=0;
      lhs(2*i+2, 5)= points[i].X;
      lhs(2*i+2, 6)= points[i].Y;
      lhs(2*i+2, 7)= points[i].Z;
      lhs(2*i+2, 8)= 1;
      lhs(2*i+2, 9)= -pixels[i].v * points[i].X;
      lhs(2*i+2, 10)= -pixels[i].v * points[i].Y;
      lhs(2*i+2, 11)= -pixels[i].v * points[i].Z;
```

```
    lhs(2*i+2, 12)= -pixels[i].v;
}

// Step 2: Find u-vector corresponding to smallest singular value (S) (=Solution)
DiagonalMatrix D(12);
Matrix U(12,12); SVD(lhs.t()*lhs, D, U); // lhs.t() denotes matrix transpose
int smallestCol = 1; // Column containing smallest sing. value
// find smallest
for (int j = 1; j < 13; j++)
    if ((D(smallestCol)*D(smallestCol)) > (D(j)*D(j))) smallestCol = j;
ColumnVector S = U.Column(smallestCol);

// Step 3: write 12x1-Vector S as 3x4 Matrix (row-wise)
Matrix ProjMV(3, 4);
for (int k = 0; k < 12; k++) ProjMV((k / 4)+1,(k%4)+1) = S(k+1);

// Step 4: decompose ProjMV in Proj- and ModelView-matrices
double scale=sqrt(ProjMV.SubMatrix(3,3,1, 3).SumSquare());
ProjMV /= scale;
//ProjMV /= ProjMV(3,4);
if (ProjMV(3,4) > 0) ProjMV *= -1;
ColumnVector Q1 = (ProjMV.SubMatrix(1,1,1,3)).t(); ColumnVector Q2 =
(ProjMV.SubMatrix(2,2,1,3)).t();
ColumnVector Q3 = (ProjMV.SubMatrix(3,3,1,3)).t();
double q14 = ProjMV(1,4);
double q24 = ProjMV(2,4);
double q34 = ProjMV(3,4);
double tz = q34;
double tzeps = 1;
if (tz > 0) tzeps = -1;
tz = tzeps*q34; RowVector r3 = tzeps*Q3.t();
double u0 = (Q1.t()*Q3).AsScalar();
double v0 = (Q2.t()*Q3).AsScalar();
double a = crossNorm(Q1,Q3);
double b = crossNorm(Q2,Q3); RowVector r1 = tzeps*(Q1.t() -
(u0*Q3.t()))/a; RowVector r2 = tzeps*(Q2.t() - (v0*Q3.t()))/b;
double tx = tzeps*(q14 - u0*tz)/a;
double ty = tzeps*(q24 - v0*tz)/b;
// create Rotation Matrix and Translation Vector
Matrix RotMatrix(3,3); RotMatrix = r1 & r2 & r3;
ColumnVector t(3);
t << tx << ty << tz;
```

// Step 5: Expand found matrices to 4x4 matrices

```
// Projection
Matrix IntMat(4,4);
IntMat(1,1)=-a; IntMat(1,2)=0; IntMat(1,3)=-u0; IntMat(1,4)=0;
IntMat(2,1)=0; IntMat(2,2)=-b; IntMat(2,3)= -v0; IntMat(2,4)=0;
IntMat(3,1)=0; IntMat(3,2)=0;
IntMat(3,3)=-(gfFarPlane+gfNearPlane)/(gfFarPlane-gfNearPlane);
IntMat(3,4)=-2*gfFarPlane*gfNearPlane/(gfFarPlane - gfNearPlane);
IntMat(4,1)=0; IntMat(4,2)=0; IntMat(4,3)=-1; IntMat(4,4)=0;
// Rotation&Translation
Matrix ExtMat(4,4); RowVector nulleins(4);
nulleins << 0.0 << 0.0 << 0.0 << 1.0;
ExtMat = ((r1 & r2 & r3) | t) & nulleins;
```

// Step 6: Set matrices as Current MV/Proj-matrices

```
for (int l = 0; l < 16; l++) {
    MVMat[l] = ExtMat((l%4)+1, (l/4)+1);
    ProjMat[l] = IntMat((l%4)+1, (l/4)+1);
}
// ProjMat has to be multiplied by VP^-1 mat16dMult( invViewport, ProjMat, ProjMat);
```

// Step 7: Save matrices to file ("IntMat0.dat"/"ExtMat0.dat")

```
ofstream lfInt, lfExt;
lfInt.open("IntMat0.dat", ios::out);
```

```
  lfExt.open("ExtMat0.dat", ios::out);
  for (i = 0; i < 16; i++) {
        lfInt << IntMat(i%4 + 1, i/4 + 1) << ((i%4-3)?" ":"\n");
        lfExt << ExtMat(i%4 + 1, i/4 + 1) << ((i%4-3)?" ":"\n");
  }
}
```