TECHNISCHE UNIVERSITÄT WIEN

DISSERTATION

The Personal Interaction Panel
a two-handed Interface for Augmented Reality

ausgeführt zum Zwecke der Erlangung des akademischen Grades eines Doktors der technischen Wissenschaften unter Leitung von

Ao. Univ. Prof. Dipl-Ing. Dr.techn. Michael Gervautz
E 186 / 2
Institut für Computergraphik

eingereicht an der Technischen Universität Wien
Technisch-Naturwissenschaftliche Fakultät

von

Dipl.-Ing. SZALAVÁRI Zsolt
Matr. Nr. 9326205
Hernalser Hauptstraße 168 / 18, A-1170 Wien

Wien, im September 1999

Abstract

The way we perceive the rapidly expanding information environment will change dramatically over the next years. Augmented Reality research offers a smooth immersion into this parallel world, by overlaying spatially aligned three-dimensional computer-generated information onto a human's view. However, to manipulate the perceived information new types of interfaces are needed that are smoothly integrated at the border of our real world and the information space.

For the interaction with this virtual content the Personal Interaction Panel (PIP), a two-handed interface for Augmented Reality is introduced. The underlying design idea employs existing human skills for the interaction in a new environment, so that the interaction task itself does not induce additional cognitive load on the user. The careful conception of the PIP interface ensures that it conforms to previous results in bimanual action and benefits from the observations of this research field.

The results of the analysis of existing methods to interact in Augmented Reality are summarized as Basic Design Guidelines. These guidelines are the basis for the theoretical formulation and conception of the proposed interface. A wide spectrum on different metaphors is presented that benefit from the underlying interface design and thus improves the user's interaction abilities.

The implementation of the described theoretical research and the field-testing in a number of application scenarios helped to prove the described ideas. Using the experience gained from this work, the interface was applied to different environments with divers demands. The influences from these experiments helped to recognize the universal character of the Personal Interaction Panel, and to conclude that the PIP interface can be a general tool for interacting with virtual content in many different environments.

Kurzfassung der Dissertation

Die Art und Weise, wie wir mit dem ständig wachsendem Informationsraum, der uns umgibt umgehen wird sich in den nächsten Jahren dramatisch ändern. Die Forschung auf dem Gebiet der Augmented Reality bietet eine sanfte Methode sich dieser parallelen Welt zu nähern. Sie überlagert den menschlichen Blick mit dreidimensionaler computer-generierter Information, die räumlichen Bezug hat. Um diesen wahrgenommenen Informationsraum beeinflussen zu können werden neue Schnittstellen gebraucht, die an der Grenze zwischen realer- und virtueller Welt angesiedelt sind.

Diese Arbeit trägt zu dieser Problemstellung mit der Einführung des Personal Interaction Panels bei. Dem zweihändigen Interface liegt die Idee zu Grunde, dass existierende menschliche Fähigkeiten verwendet werden, um auf eine neue Umgebung einzuwirken. Ziel ist auch, dass die Manipulation selbst keine kognitive Belastung darstellt, sodass sich der Benutzer auf die eigentliche Aufgabe konzentrieren kann. Das sorgfältige Konzept sichert, dass die vorgestellte Schnittstelle vorangegangenen Erkenntnissen aus dem Bereich der beidhändigen Interaktion entspricht und von dessen Resultaten profitiert.

Die Ergebnisse der Untersuchung anderer Interaktionsmethoden in Augmented Reality Umgebungen wurden als Leitfaden zum Entwurf neuer Schnittstellen zusammengefasst. Diese Richtlinien liegt auch der theoretischen Formulierung und Ausarbeitung der vorgestellten Schnittstelle zu Grunde. Ein breites Spektrum an Anwendungsmöglichkeiten wird vorgestellt. Diese profitieren vom Interfacedesign und erweitern so die Interaktionsfähigkeiten des Benutzers.

Die beschriebenen Ideen konnten mit Hilfe von verschiedenen Implementierungen und der Erprobung im praktischen Einsatz belegt werden. Mit Hilfe der gewonnenen Erfahrungen konnte die Schnittstelle auch in anderen virtuellen Umgebungen eingesetzt werden. Die Erkenntnisse dieser Experimente verhalfen zum Erkennen der Universalität des Personal Interaction Panels. Daraus resultierend wurde festgestellt, dass die PIP-Schnittstelle als allgemeines Werkzeug für die Handhabung von virtuellen Inhalten in verschiedensten Szenarien betrachtet werden kann.

Acknowledgments

This page is dedicated to all people helping this thesis to its existence:

First I want to express my gratitude to my supervisor *Michael Gervautz*, who supported my thesis in all possible ways and inspired me to this imagination. I like to thank *Werner Purgathofer* for adopting me to his group in Vienna.

As a cohabitant of Studierstube I thank *Dieter Schmalstieg* and *Anton Fuhrmann. Dieter* motivated me in several discussions and opened our eyes for transparency. *Toni* set-up our common snuggery in every mean.

I also want to thank my colleagues *Erik Eckstein, Reinhard Sainitzer, Herbert Buchegger, Christian Faisstnauer,* and the students *Hermann Wurnig* and *Wolfgang Greimel* for their great work in implementing PIP applications. I render thanks to *Robert Tobler*, who gave me his digital Nikon to make some of the photos for this thesis and *Helwig Löffelmann* for continuously picking at me to write down my thesis.

I particularly thank my *parents* for growing me up in love and to ground me in struggling.

Last but not least my special thanks go to my wife *Veronika* for her patience with me and my little *Zsófi*, whom I missed to see when standing up the first time, because of finishing this thesis – I hope you will forgive me!

Table of Contents

Thot – Egyptian god of writing and science.
(~ 3000 BC)

Chapter One - Introduction

Science heads the goal to describe the universe as detailed as possible in every discipline, to open the *known universe* at each scale. Even though some philosophic groups deny even the existence of a universe, the rest of mankind always wanted to amplify the own senses to get behind the directly perceived reality. Examples of methods for this amplification of the sensed reality reach from the taking of hallucinogens, the invention of the telescope, the design of spectrometers to the huge accelerators for picking the building blocks of matter into pieces. Common in these approaches is the effort to *show* the invisible part of reality, which was not present to the senses before.

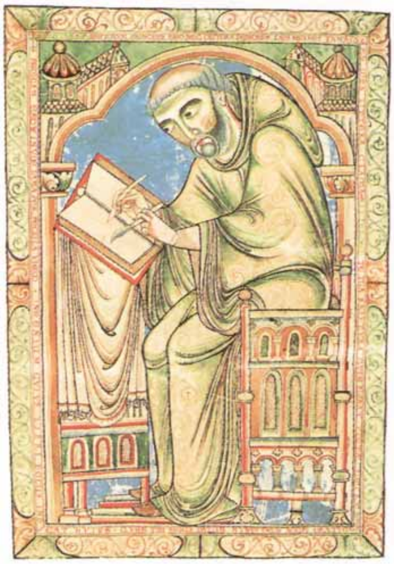
To *show* an invisible part of surroundings is also the most important objective of the method called *Augmented Reality* (AR).

This method for *visual improvement or enrichment of the surrounding environment by overlaying spatially aligned three-dimensional computer-generated information onto a human's view* presents a synthetic reality designed to amplify our perception. Artificially generated objects or any kind of abstract information appears to the observer blended with the everyday reality in three dimensions.

Why is AR an interesting topic and will be a useful tool in the future? The ability to represent not directly perceptible information with virtual objects makes tasks easier to perform for humans. This visual computer support is a specific example of what Fred Brooks describes in general as *intelligence amplification* [Brooks, 1996].

More than just simple visual enhancement and support, interactive augmented reality applications will change the way of interaction with abstract information in future. By letting users of AR systems to manipulate augmented content, the blend of real and virtual becomes yet more effective as flexibility is introduced into the system. This powerful tool has a great potential for a broad range of applications, including mobile context-sensitive information systems, scientific visualization, in-place display of measurement data, medicine and surgical planning, education, training and entertainment.

To come up to the application expectations user interaction with the supporting computer has to undergo dramatic changes compared to today's input devices, which concentrate on a text based input or two-dimensional selection. We are at the beginning of a new era of interaction paradigms to handle three-dimensional content in a natural and flexible way. Present work contributes to this evolution with the introduction of the *Personal Interaction Panel* – the *PIP* –, a two-handed interface for augmented reality.



Eadwin – Monk at the Canterbury Abbey.

Chapter Two - State Of the Art

2.1 *Introduction*

Since the beginning of time mankind was always fond of using images to express various thoughts. Carved into the wall of a cave, painted onto papyrus or simply described with words to form a mental image. Partners in a discussion envision ideas or plans to share a common knowledge on an abstract topic. Communicating theoretical content using images makes comprehension much faster as humans acquire the most information through their eyes.

The information age created powerful tools to create, transport and show images. Static pictures, figures, and photographs evolved into moving pictures and animation. Stored earlier on celluloid, nowadays on magnetic media or optical discs and distributed via radio waves and cables, dynamic images are a basic and integral part of our everyday life. Now we are about to cross the next frontier of interactive images. Multimedia and random access video media and systems allow bypassing the linearity of time and history of a moving picture. Each time experienced - the content is presented in a different and new way. Learning and understanding break with tradition, from now on acquiring knowledge becomes a non-linear process of interactive information exploration.

The next frontier to cross will be to explore, create, and communicate information directly in three dimensions. The experience will be interactive in a comfortable and natural way. Reality and synthetic content will be seamlessly integrated in a mixture that amplifies human intelligence. To work with this synthetic content completely new interfaces are needed. In fact, interfaces or interaction devices will be fundamental part of these augmented environments.

Where are we now? How far is this next frontier and what can today's technology provide to make the first small steps towards this future? The next section tries to summarize this and give an overview of this field.

2.2 *Survey of Augmented Reality*

2.2.1 Virtual Reality and Augmented Reality

As a highly interdisciplinary field, scientific visualization frequently requires experts with different background to cooperate. Collaborators may have different preferences concerning the chosen visual representation of the data, or they may be interested in different aspects. An efficient collaboration requires that each of the researchers have a customized view of the data set. At the same time, presence in the same room is preferred because of the natural interaction during a discussion. These requirements can uniquely be fulfilled in an augmented reality system that combines real world experience of the collaborators and physical equipment with the visualization of the synthetic data.

Why is AR Different?

Compared to visualization in immersive virtual reality, augmented reality allows the use of detailed physical models, the properties of which cannot be met by their virtual counterparts: arbitrarily detailed visual representation, no visual or temporal artifacts and force-feedback for free. Only those aspects of the model that cannot be seen in reality have to be added by the computer system. For example, one could take the physical model of an airplane or airplane wing to investigate the flow around this object, which is simulated by computer and added to the display. Manipulation of the real world model (e. g. its orientation) is more intuitive and simpler to support than a purely virtual environment. A related example would be the use of a humanoid torso or puppet that is overlaid with medical information from inside the human body in the style of [Bajura, 1992].

This combination of conventional experimental work with scientific visualization and augmented reality technology leads to the concept of an augmented laboratory, which would provide a superior research environment in which to conduct experiments that are executed solely inside the computer, while maintaining a conventional and familiar work setup.

The *Studierstube* approach concentrates on the seamless combination of a physical world workspace and an augmented environment for multiple users in three dimensions, with unaffected social communication channels and an augmented user interface that supports natural handling of complex data at interactive rates. In this type of distributed multi-user systems adequate communication strategies for continuous synchronization and real-time performance are required that also allow the interaction with a shared geometric database. *Studierstube* is described in more detail in chapter 2.2.2.

Despite the extraordinary rapid development of computers and software for virtual reality, the acceptance of full immersive systems follows these growth rates rather hard. The aim to create the feeling of immersion by presenting convincing stimuli to the user is still not really satisfying, we are far from high-fidelity in virtual reality. Especially social aspects of full immersion may play a substantial role in denying usage by some people.

Augmented reality offers a smooth immersion by leaving the connection to real-world environments basically untouched and superimposing computer generated imagery onto real artifacts. Social communication channels as natural speech and paralanguage are not blocked, breaking down mental barriers of applying virtual reality technology to a specific problem. Not unexpectedly, a lot of different problems arise in the investigation of augmented reality, like registration and occlusion of real world obstacles, but much research is concentrating on these topics e.g. [Bajura, 1995] and [State, 1996a]. Tracking precision and tracking range in larger environments, especially for outside applications are still unsatisfying and need further investigation, however some solutions are promising, like presented in [Auer, 1999] and [Feiner, 1997].

2.2.2 Augmented Reality Systems

Evolution

The evolution of augmented reality started in the early days of computer graphics, when Sutherland pioneered research on head-mounted displays [Sutherland, 1968]. His work still inspires the virtual reality research community of today. Although only capable of simple vector drawings, his prototype head mounted display was the first binocular see-through system, effectively the first augmented reality system.

Feiner et al. described in [Feiner, 1992] and [Feiner, 1993] a knowledge based augmented reality system. As a demonstration, they choose to configure the system to support people with the maintenance of laser printers (Figure 16). However, a lot of effort is required to generate accurate models and extremely precise registration. Bajura et. al. [Bajura, 1992] described a medical visualization system based on augmented reality techniques. A see-through head mounted display (HMD), also developed at UNC [Holmgren, 1992], allows geometrically correct superposition of ultrasound data of the unborn onto the belly of the mother-to-be, so the gynecologist can examine the position of the unborn within the mother. Another medical application of AR has been presented by State et. al. [State, 1996b] for ultrasound guided needle biopsy of the breast. Sharma and Molineros [Sharma, 1996] present a system for mechanical assembly guidance using annotations attached to real world scenery.

Scientific visualization in virtual reality becomes increasingly a field of interest for many researchers. In the early 90ies the GROPE project-group around Fred Brooks produced a haptic arm-like device and a large stereo display for the visualization and manipulation of chemical data [Brooks, 1990]. Their *nanoManipulator* [Taylor, 1993] (see also Figure 13) allows precise manipulation of a scanning tunneling microscope and works also with force feedback. Another important milestone for the combination of VR and scientific visualization was the development of the virtual wind tunnel at NASA-AMES by Steve Bryson. Using a *Boom*-device and a data glove as interaction tool [Bryson, 1991], scientists were able to see and interact with true stereoscopic images of a flow field visualization. A follow-up project, the distributed wind tunnel [Bryson, 1993] was developed, which divided computation in a distributed system for better efficiency, and allowed multiple users to experience the simulation at the same time. Collaboration in a distributed virtual environment, not necessarily limited to scientific visualization has been proposed by Fahlén et. al. [Fahlén, 1993].

Categorization

From the *design principles* point of view today's augmented reality systems can be categorized into two major groups. Depending on the applied technology we differentiate between *optical vs. video based AR systems*. Each has particular advantages and disadvantages. Both design principles have their specific application fields, where decision is easy to accomplish. However, in many cases one has to trade in some benefit of a system for a drawback of the other to gain maximum suitability.

From the *display devices* point of view we can differentiate between *head-mounted displays*, *hand-held displays*, *monitors*, and *projection devices*. Some researchers define augmented reality in combination with head-mounted displays. The following example shows that a definition fixed to one display system does not cover all fields of this evolving technology. The group of *hand-held display based AR systems* represents a very interesting domain for the next generation of hand-held devices like palmtop computers, personal digital assistants (PDAs), and mobile communication devices. Fritzmaurice and Rekimoto presented in [Fritzmaurice, 1993] and [Rekimoto, 1995] systems which augment real surroundings with additional information based on hand-held displays.

Table 1 shows an overview of existing augmented reality systems. Based on the above categorization systems can be identified with one row/column combination. Additional to true augmented systems the table also contains VR systems, namely *CAVE* and the *Responsive Workbench*, that do not augment reality but provide an experience for a group of users. This multi-user feature as third dimension of Table 1 is discussed below in comparison to multi-user AR systems like *Studierstube*.

Display device	Design principle		
	video based AR system	Optical see-through AR system	multi-user VR system
head-mounted display	[Bajura, 1992], [Auer, 1998]	[Feiner, 1997], [Schmalstieg, 1996], [Fuhrmann, 1997], [Szalavári, 1998a]	
hand-held display	[Fritzmaurice, 1993], [Rekimoto, 1995]		
monitor	[Peuchot, 1995], [Serra, 1995]		[Carlsson, 1993]
projection	[Darell, 1994], [Petta, 1999]	[KEO, 1999], [Schmalstieg, 1999a]	[Krüger, 1995], [Cruz-Neira, 1993a]

Table 1. Categorization of Augmented Reality systems.

Video-based AR systems combine real and virtual image of an augmented environment by using cameras for the user's view of the real world (Figure 1) from [Azuma, 1997]. Combination of both images is completed in the last step of the *rendering pipeline* of the underlying computer system. This helps to correct timing lags, gives full control over the transparency of a virtual object, the contrast ratio of both images and the acquired video image can be used for additional tracking strategies. A list of video-based AR systems can be found in [Azuma, 1997]. [Edwards, 1993] gives a description for modifying an HMD to a video see-trough HMD.

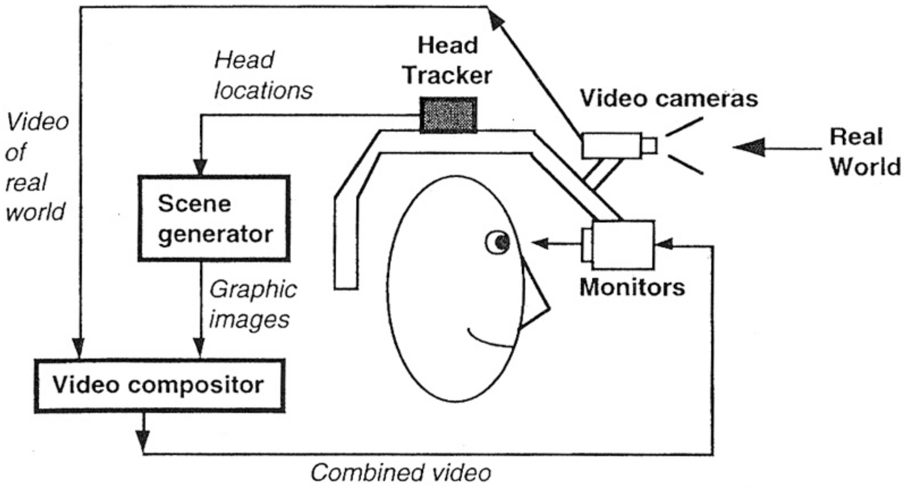


Figure 1. Conceptual diagram of video-based augmented reality systems [Azuma, 1997].

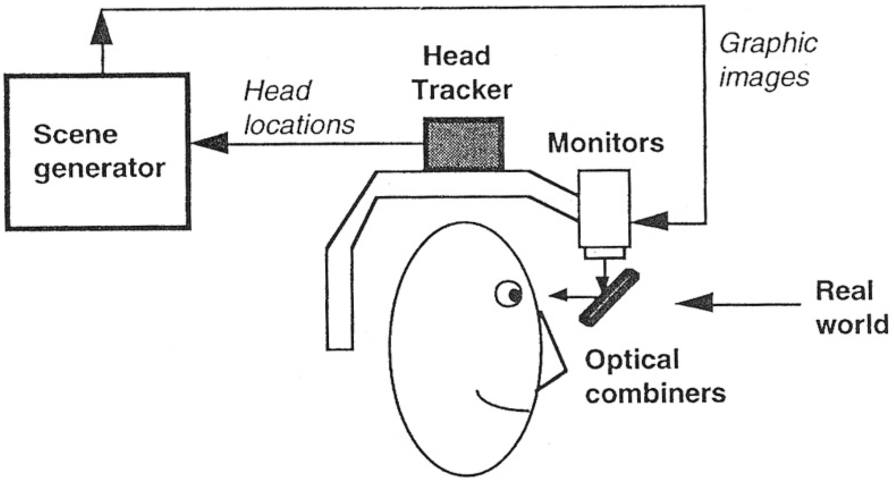


Figure 2. Conceptual diagram of see-through augmented reality systems [Azuma, 1997].

Optical see-through AR systems utilize special optics to combine the images of real surroundings reaching the eye with computer generated overlays (Figure 2). These systems tend to be much simpler in design compared to video based solutions. Advantages of such systems are simplicity and safety in concern of system breakdowns. Furthermore the real image keeps the original (high) resolution and no eye offset has to be taken care of as in video based solutions. Up to the present day all optical see-through systems employ see-through HMDs like described in [Szalavári, 1997], [Billinghurst, 1996], [Feiner, 1992], and [Feiner, 1993], though these systems depend mainly on the availability of these devices. Currently SONY's Glasstron [Glasstron, 1999] series offers a good price/performance ratio for see-through HMD applications. Schmalstieg presents in [Schmalstieg, 1999a] the idea of augmented VR, as a mixture of augmented- and virtual environments.

Multi-user systems

Most existing augmented applications are single user setups, or do not exploit the multi-user character of their systems. Some of the few exceptions are described here in detail as present work contributes to these set-ups: the *CAVE* system, the *Responsive Workbench*, the *TransVision* setup, the *Shared Space* and *Studierstube*.

In the *CAVE* system [Cruz-Neira, 1993a] and [Cruz-Neira, 1993b] users see stereoscopic 3-D scenes with LCD-shutter glasses on large projection walls surrounding them (Figure 3). Position and orientation of one user's head is tracked, so that the images on all walls correspond to this viewer's position. The viewers have the impression to be surrounded by a 3-D virtual scene. However the CAVE is not an AR setup in the common sense, due to its size it can incorporate real mock-ups and thus mix reality and virtuality. A disadvantage of this system is that the presented images fit to the head position only for one viewer; noticeable visual artifacts exist for all other viewers. Interaction with the application can take place in two different ways. First the CAVE system has in most installations a control desk, where an operator can control any application interface from a console. Inside the CAVE a *wand* as shown in Figure 4 is used to control viewpoint and to accomplish application specific control tasks. This wand is a handle with 3 buttons and is magnetically tracked in position and orientation.

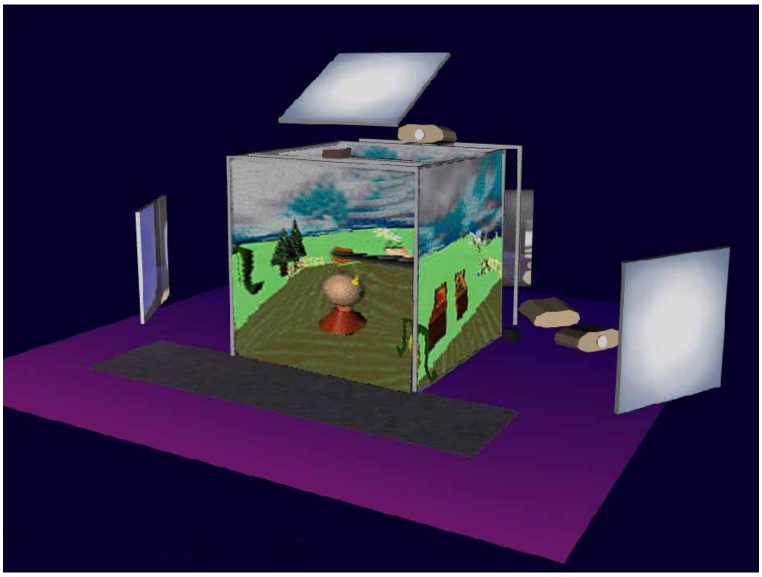


Figure 3. The CAVE system. The setup figure shows the projection devices and gives an impression of the necessary effort for the installation of this system.

Figure 4. The *wand* used in most CAVE installations.

The *Responsive Workbench* [Krüger, 1995] uses one display area, which is built into a tabletop. A CRT projector is projecting the computer-generated images from behind the projection screen. Like in the CAVE, viewers wear LCD shutter glasses and only two users can see the objects in correct stereoscopy at a maximum as presented in [Agrawala, 1997]. Furthermore, a relatively steep viewing angle is necessary to achieve a good 3D impression, i.e. the viewers have to stay close to the table. Very similar systems have been presented of different groups in tight cooperation with manufacturers of these systems. *ImmersaDesk* [Pyramid, 1999], *Immersive Workbench* [FakeSpace, 1999a], *Baron Table* [Barco, 1997] (Figure 5) trademarks cover the same metaphor for a table mounted projection systems with slight differences in tilting of the table or the mobility of the whole apparatus. However, no interaction device has been directly mated with table like display systems, most applications use so far a *CyberGlove* [Virtex, 1999] or *PINCH Gloves* [FakeSpace, 1999b]. A modified transparent version of the *Personal Interaction Panel* has been successfully implemented and tested with an urban design application as presented later in chapter 6.3.

Figure 5. The *Baron Table*, a table-like display system.

Figure 6. The *TransVision* system in action while to scientists interact with virtual objects.

Common to both presented systems is that image generation is performed by one display system for all participants of the experience. This fact limits the range of applications to a subset, where correct stereoscopic view for all participants is not crucial. To accomplish this Rekimoto presented in [Rekimoto, 1996] the *TransVision* system a multi-user setup using hand-held displays (Figure 6). In this system image generation moved from one common display unit to personal palmtop screens, however the presented augmentation is only a monoscopic image. An interesting feature of this system is that the display device itself represents an interaction device. Chapter 2.3.2 describes this interaction metaphor in more detail.

Closest to our work *Studierstube* system is the prototype implementation of *Shared Space* [Billinghurst, 1996] shown in Figure 7. Users wearing head-mounted see-through displays can discuss shared information in three dimensions floating around them in space and interact using gestures and speech commands. As the focus of this work is on ubiquitous computing and not in situ cooperative work, distribution of data, information sharing and interaction techniques face different problems as presented in our work.

Figure 7. Two users communicate using the *Shared Space* system.

Studierstube

The Institute of Computer Graphics of the Vienna University of Technology develops in cooperation with the Computer Vision Group of the Graz University of Technology a multi-user augmented reality system called *Studierstube*. The project was name selected, after the play *Faust* by Johann Wolfgang von Goethe, in which the leading character uses a study room for performing research and philosophy: the *Studierstube* (Figure 8).

Daß ich erkenne, was die Welt
Im Innersten zusammenhält,
Schau alle Wirkenskraft und Samen,
Und tu nicht mehr in Worten kramen.

To realize what holds the world
Together in its core,
I see all seeds and force of act
And search for words no more.

Johann Wolfgang von Goethe, Faust

Figure 8. *Studierstube* – unveiling hidden secrets.

In [Gervautz, 1996], [Schmalstieg, 1996], [Fuhrmann, 1997] and more detailed in [Szalavári, 1998a] we proposed a system capable of visualization of three-dimensional scientific data for multiple simultaneous viewers within one room in contrast to outside AR applications. The choice of this setting limits the complexity of the problem, as the "real world" is limited to a room, which is complemented by the "virtual world". Each viewer wears magnetically tracked see-through HMDs providing a stereoscopic real-time display, and can freely walk around in order to observe the augmented environment from different viewpoints (Figure 9).

Figure 9. *Studierstube* in a conceptual draft. – Three people wearing see-trough glasses at a meeting, viewing a virtual globe. Note that the table is an object in the real world, the globe just an image overlaid on the real surroundings by the headset.

The mixture between real and virtual visual experience, created in our system by see-trough HMDs, is a key feature of our system. Thus, it is possible to move around freely without fear to bump into obstacles, as opposed to fully immersive displays, where only virtual objects can be perceived. This enables a work group to discuss the viewed object, because the participants are seeing one another and can therefore communicate in the usual way (Figure 10).

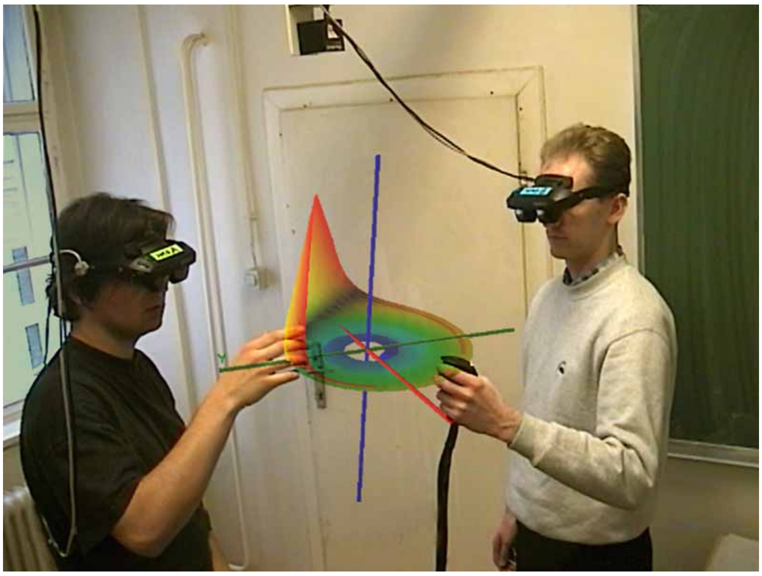


Figure 10. Two users investigate the Rössler-attractor in *Studierstube.*

The following key properties summarize the attributes of the system:

◊ *Virtuality* – Viewing and examining of objects that are not accessible directly or that do not exist in the real world can be carried out in this environment.

◊ *Augmentation* – Real-world objects can be augmented with spatially aligned information. This allows smooth extension of real objects with virtual properties in design processes, like variations of new parts for an existing system.

◊ *Multi-user support* – Much research has been devoted to the question how conventional software and desktop computers can be enhanced with measures to support effective group interaction. Fortunately, a benefit of augmented reality is that sophisticated groupware mechanisms are not really needed to perform real work. Normal human interactions (verbal, gestures, etc.) are easily possible in an augmented reality setup, and they are probably richer than any computer-governed interaction can ever be.

◊ *Independence* – Unlike the *CAVE* and the *Responsive Workbench*, control is not limited to a guiding person, while other users act as passive observers. Each user has the option to move freely and independently of the other users. In particular, each user may freely choose a viewpoint with stereoscopy for correct depth perception. But not only is observation independent, interaction can also be performed on a personal base.

◊ *Sharing vs. Individuality* – Investigated objects are in general shared among users, in the sense of visibility, this means that all participants can see the same coherent model, consistent in its state over time. By presenting the visual sensation directly to each user with the lightweight see-trough HMDs, the displayed data set can also be different for each viewer, as required by the application's needs and the individual's choice. Personal preferences on different layers of information can be switched on and off, as known from CAD packages. A unique *SEAM* mechanism [Schmalstieg, 1999b] enables interactive revealing and hiding of information in different layers in three dimensions.

◊ *Interaction and Interactivity* – With the support of augmented tools like the proposed *Personal Interaction Panel*, visualized data can be explored interactively. Changes inherent in the scientific simulation can be viewed immediately.

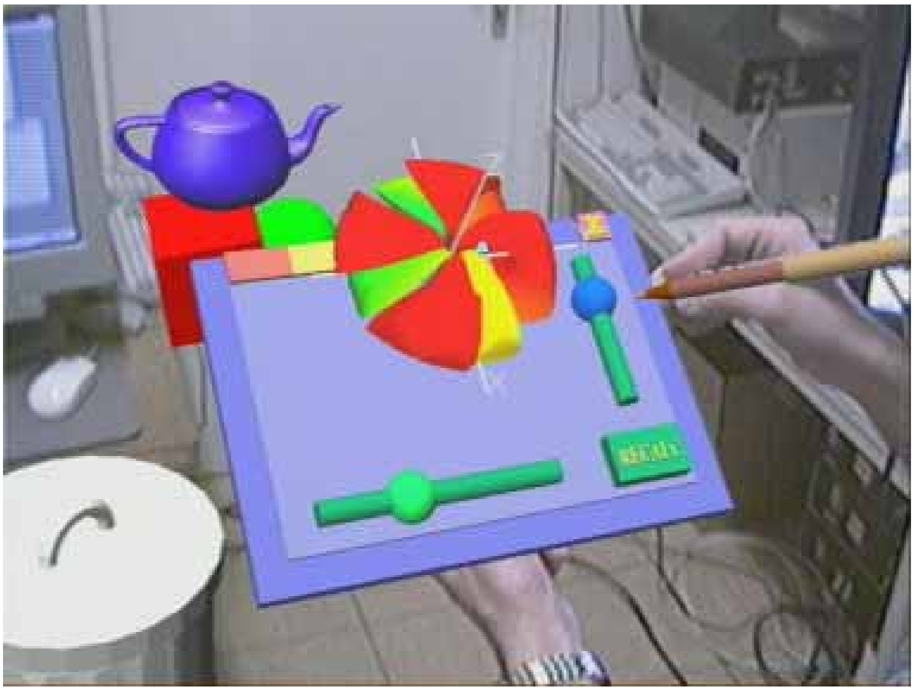


Figure 11. PIP application in Studierstube.

Current work focuses on the design of the *Personal Interaction Panel* (Figure 11). However, this design process is tightly connected to the development of *Studierstube* at the Institute of Computer Graphics of Vienna University of Technology. Therefore throughout this work *Studierstube* will be often referenced as implementation platform or test-bed for PIP features and applications.

2.3 *Interaction in Augmented Reality*

2.3.1 Interacting with Information

In real environments people are used to interact with objects. Our everyday experience is to grab things, manipulate, and put them away. In the last decades "non-real" elements populate our environment, we interact more and more with pure information. Computers are currently the most widespread way to interact directly with information. Sadly not computers did evolve to interface humans as best as possible.

Figure 12. Collection of traditional 2D input devices (mouse, keyboard, and tablet).

Engineers developed at the current level of technology devices and metaphors that were easy to understand and cheap to implement in masses (Figure 12). Most of these devices had one in common: They more or less mainly supported the technical needs of computers. Users had to adapt to the input devices and the interaction metaphors.

New computer generations incorporating graphics hardware allowed the visualization of three-dimensional problems at interactive rates. The introduction of Stereographics's Crystal Eyes glasses in 1980 for 3D output on workstation monitors allowed seeing these visualizations even in stereoscopic 3D for a broad community. Three-dimensional applications emerging on these machines required a new way of interaction with the computer. Researcher first employed existing 2D hardware interfaces to implement new interaction metaphors. [Chen, 1988] and [Nielson, 1986] give an example of using 2D devices for 3D rotation and manipulation. However mapping 2D input to 3D actions plays even today a significant role in Desktop-VR applications, this type of interaction lacks direct correspondence of real and virtual space.

Removing the necessity of mental abstraction of space a new generation of input devices was created. These were mainly constructed around specific tasks like 3D modeling, scene-composition, layout planing, etc. Because of this, a wide variety of completely different 3D input devices have been introduced. Chris Hand gives in [Hand, 1995] and [Hand, 1997] a good overview on this development of 3D input devices from the early beginnings in mid 80's.

Whereas conventional desktop input devices, e.g. keyboard and mouse have reached a high degree of specialization during a synthetic evolutionary process, 3D input devices have still significant disadvantages. High accuracy mechanical devices are somewhat bulky or bound to certain applications and do not support generalized interaction techniques [Taylor, 1993] (Figure 13). Six degree-of-freedom (DOF) mice like the Spaceball 4000 FLX [Labtec, 1999] in Figure 14 and data gloves [VTI, 1999], [Nintendo, 1989] in Figure 15 extend the possible set of interactions by adding nearly unconstrained three dimensional movement and capturing dozens of position and orientation data. However, these devices suffer either from unsatisfying low interaction bandwidth or an overloaded metaphor like complex gesture languages [Sturman, 1994]. Since no direct tactile feedback of the virtual objects is provided in most systems, inexperienced users feel disoriented and find it rather difficult to work with flying buttons, menus, “3D widgets” [Conner, 1992] and similar metaphors floating around them. Finally, Pierce et. al. report in [Pierce, 1997] on image-plane interaction techniques for immersive virtual environments and let users interact with 2D projections of 3D objects, an approach that supports manipulation of distant objects in VE.

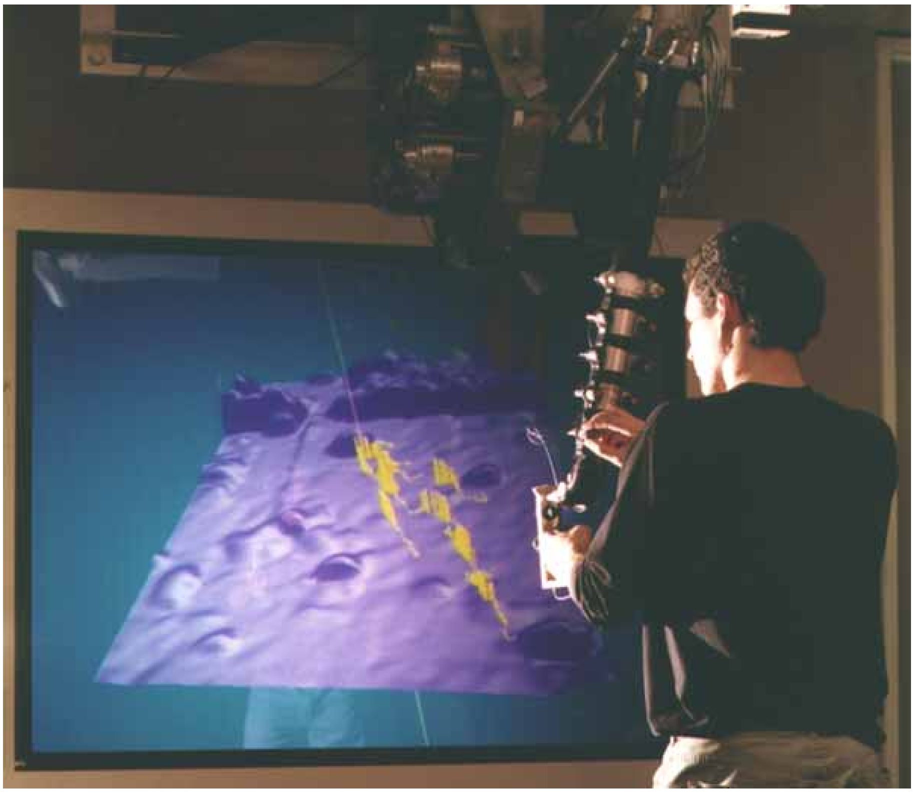


Figure 13. The *nanoManipulator*.

Figure 14. *Spaceball 4000 FLX* 6 degree of freedom input device.

Figure 15. *VTI CyberGlove* and *Mattel/Nintendo Powerglove*.

To overcome many drawbacks of other devices Wloka presented in [Wloka, 1995] the *Virtual Tricorder*. This device resembles a joystick like handle with buttons on the top. The tricorder is represented in the VR-system with a visual clone making it easy to identify. Additionally the visual similarity helps to suspend disbelief. Application control is easy to fulfil with a menu appearing at the tip of the tricorder. This menu is attached to the input device, so target acquisition is only relative, enhancing the performance of interaction. Flying menus are bound to a physical prop, making them easy to use.

A selection process should now follow this early phase of evolution, which generated an enormous number of different approaches. Highly specialized tools perform possibly much better in certain application scenarios, but fail in more general tasks.

However this might keep these metaphors alive in their application field, it hinders their general acceptance. Generalization was often disapproved at the beginning of the evolution process of input devices [Buxton, 1990] blocking possible new and different gadgets. Today we can look back and tell that many developments did not survived the struggle for acceptance. A few hardware vendors like Polhemus, Logitech and VPL licensed the rather successful devices, other remained prototypes or custom solutions forever. We can learn from their design flaws, extract the good ideas and deduct generalized interfaces like the proposed Personal Interaction Panel.

2.3.2 Interaction Metaphors for Augmented Reality

Interaction metaphors for Augmented Reality systems went through a similar process of development. However the overall number of interfaces is much smaller than in common VR applications, due to the diversity of AR systems in general interfaces are distributed along a broader spectrum.

Many early AR applications focussed on the augmentation of real environments with additional *passive* data that could be browsed in an interactive manner. Amselem [Amselem, 1995], Fitzmaurice [Fitzmaurice, 1993] and Rekimoto [Rekimoto, 1995] show one-handed interaction metaphors using a hand-held display (HHD) device for the exploration of spatially aligned databases. The magnetic tracked display can be carried around showing situated graphics augmented over the real environment. User interaction covers mainly a browsing task of large, spatially distributed multi-media database, but implementations also include features, like zooming, layers and browsing of remote environments.

Feiner's work goes a step further in interaction with both the real and the virtual part of an augmented environment. In [Feiner, 1992] and [Feiner, 1993] he describes an AR system for the maintenance of laser printers. In this system additional information is overlaid over the real part for supporting the maintenance task (Figure 16). With the ARC-system shown in Figure 17 he extends this approach for a construction task in [Webster, 1996]. Users are given to a bar-code reader for identifying parts used to construct the aluminum construction frame. Using augmented overlays the system shows 3D position of the next part to be installed. This approach does also not alter the stored information, however it modifies the displayed information step-by-step. The *Touring Machine* approach in [Feiner, 1997] extends these development for outside scenarios combining an HMD for situated information and a hand-held pen based personal digital assistant (PDA) for interaction and additional information retrieval (Figure 18). Information displayed in the HMD can be configured with the PDA. The user is also visually notified in the HMD whenever additional data is available and could be retrieved on the PDA.

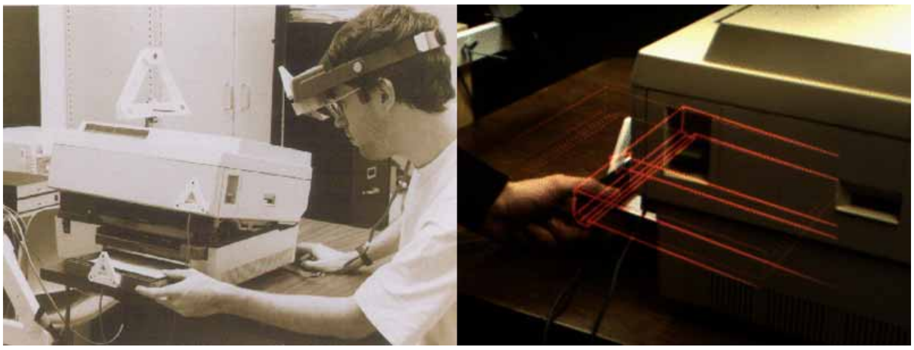


Figure 16. *KARMA* - the knowledge based AR-system provides situated visual help information.

Figure 17. *ARC* – Constructing aluminum frame systems using augmented help information. A magnetically tracked bar-code reader provides interaction with the system.

Figure 18. *MARS* – The Touring Machine allows browsing situated information in outside scenarios, e.g. the Columbia University.

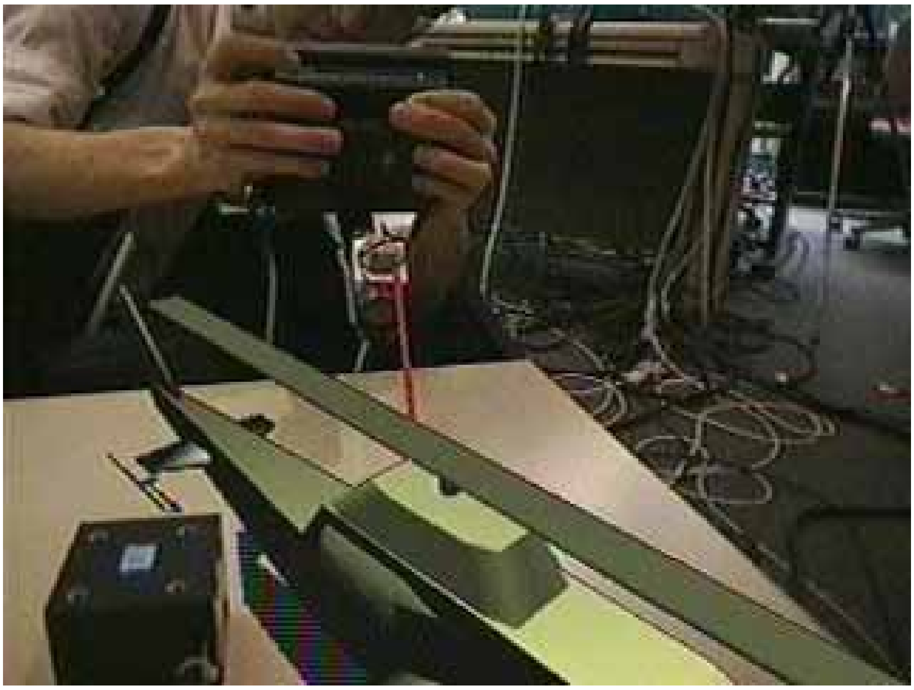


Figure 19. *TransVision* – By pushing the object selector button a beam is emitted perpendicular to the surface of the hand-held display device. The first object hit by the beam is selected for further manipualtion.

Finally, the earlier presented *TransVision* [Rekimoto, 1996] system in Figure 19 employs also interactive manipulation of 3D data in an AR-system. The display device itself represents an interaction device that is controlled with two buttons. One is used for object selection and manipulation. By pushing this button a virtual beam is emitted perpendicular to the surface of the display device. The first object hit by the beam is selected and can be manipulated. A similar approach is used in [Fritzmaurice, 1993] to make object selections. By pushing the other – menu – button a pie menu appears on the display, menu items are selected by tilting the display and releasing the button. This unique technique fits very well to this type of device, however the other hand of the user is not involved in interaction. Two-handed interaction in this context could improve the quality of interaction. To show how this can be proven, the following chapter summarizes existing work on bimanual interaction with non-real content.

2.4 *Bimanual Interaction*

2.4.1 Psychology

Two-handed input has often been viewed as a technique to improve the efficiency of human-computer interaction, by enabling the user to perform two sub-tasks in parallel [Buxton, 1986], rather than sequentially selected modes. When interacting in three dimensions, Hinckley found that using two hands not only improves efficiency, but can also help to make spatial input comprehensible to the user [Hinckley, 1994a]. Enabling the use of both hands can allow users to ground themselves in the interaction space; in essence the user's own body becomes a spatial reference. Mine exploits this ability called *proprioception* to enhance the precision of remote manipulation of objects, which are out of reach of the user [Mine, 1997]. Sachs observed an even more interesting phenomenon: "the simultaneous use of two (spatial input) sensors takes advantage of people's innate ability knowing precisely where their hands are relative to each other" [Sachs, 1991]. This ability arises from the *kinaesthetic feedback* [Hand, 1997] that allows us to know the position of our limbs relative to our body. Hinckley, Pausch and Goble documented in [Hinckley, 1994b] and [Goble, 1995] their observation with several hundred test users of a two-handed spatial interface for neurosurgical visualization. This informal study reaffirmed and strengthened Sachs's observation: most test users can operate this interface effectively within their first minute of use with little or no training at all. Findings of Buxton [Buxton, 1986] and Kabbash [Kabbash, 1994] could be reinforced that users are able to transform everyday skills for manipulating tools with two hands to human-computer interfaces. Newer studies in [Shaw, 1996], [Hinckley, 1997a] and [Zeleznik, 1997] again validated these results while performing different tests in varying software environments and application scenarios. In [Shaw, 1998] Shaw analyzed pain and fatigue in a Desktop-VR environment. Results show that a well-designed two-handed metaphor can not only be more efficient in fulfilling 3D tasks, but also – being less fatiguing than a one-handed interface – be a more comfortable human-computer interface replacing other metaphors.

Much of the above cited work roots back in Guiard's early analysis of human skilled bimanual action [Guiard, 1987]. He provides an insightful theoretical framework for hypothesizing which classes of two-handed interfaces might improve performance without inducing additional cognitive load. Guiard has proposed the following principles based on his observation of right-handed subjects:

- ◊ "Motion of the right hand typically finds its spatial references in the results of motion of the left hand." For example, when writing, the left hand controls the position and orientation of the page, while the right hand performs the actual writing by moving the pen relative to the left hand.

- ◊ The right and left hands are involved in asymmetric temporal-spatial scales of motion. In the writing task, for example, the movements of the left hand adjusting the page are low in temporal and spatial frequency compared to the high-frequency, detailed work done by the right hand.

- ◊ "The contribution of the left hand to global manual performance starts earlier than that of the right." The left hand first positions the paper, then the right hand begins to write.

Throughout this thesis the phrases *dominant hand* and *non-dominant hand* will be used from now on to describe right and left hand for right-handed subjects and similar left and right hand for left-handed subjects as accepted in human-computer literature!

Guiard's conclusions imply that bimanual interaction can rise overall performance, especially in cases where asymmetric division of labor is applied to the hands. Nevertheless the application of a bimanual interface for a specific task needs careful analysis, since the enriched interface may possibly degrade the quality of interaction in some cases, like in [Kabbash, 1994], where the two-handed metaphor complied to above guidelines, but in some cases induced an additional cognitive load.

2.4.2 Bimanual Interfaces for 3D Interaction

Long before the necessary computer hardware was really available, the visionary idea of a mobile pen based computer was published in two papers. A group of researchers at the University of Illinois described in [Mel, 1988] their concept of *Tablet* – the personal computer in the year 2000. It is a notebook sized LCD display with pen based input, incorporating wireless communication over infrared and radio wave connections. The authors predict that by the year 2000 this device could be a companion throughout our everyday life.

Mark Weiser shows in [Weiser, 1991] some years later prototypes of a similar concept developed at Xerox Palo Alto Research Center. Although the prediction did not hold in every detail, most of it was realized when looking at commercially available *Personal Digital Assistants* (PDAs) like the earlier Apple's Newton Message Pad or 3Com's Palm VII (Figure 20).

Where is the connection to augmented reality one might ask? These early visions of two-handed devices influenced the development of many following interfaces! Using AR technology many aspects of the ideas that were set at that time could be achieved in the last decade. But research even exceeded all expectations like the following examples and current work show.

Sachs et. al. carried out an early example of bimanual 3D-interaction research in non-immersive desktop applications. They show in their paper *3-Draw* [Sachs, 1991] how insufficient sophisticated CAD tools serve shape design. Being unable to rough out initial ideas directly with these interaction techniques, designers prefer to use pen and paper instead. A sufficiently intuitive and easy to use approach of "design directly in 3D", using two hand-held six-degree-of-freedom sensors in form of a stylus and palette has been proposed (Figure 21). The palette is used to define a reference frame to which objects being drawn are attached, giving the possibility of moving the object instantly into the right angle for viewing on the monitor. While giving the palette a secondary problem, the pen is employed to draw and edit free-form curves directly in 3D. However, the system was not an immersive and head-tracked application, it was suitable for CAD shape design.

Figure 20. Scratchpads and PDA. The left picture shows a concept prototype model from Xerox PARC in 1991. The right picture shows 3Com's Palm VII available for 599, - USD in major computer stores in 1999.

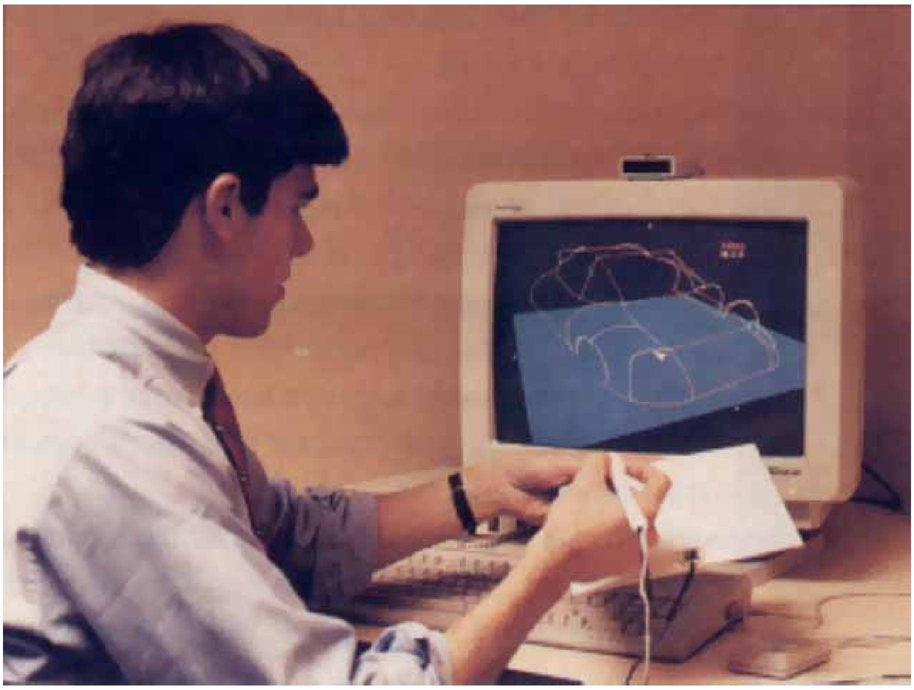


Figure 21. Designer sketching automobile parts with the *3-Draw* interface, an early example of two-handed 3D-interaction device for desktop applications.

Goble and a group of researchers at the University of Virginia developed a highly specialized two-handed interface for neurosurgical planning [Goble, 1995]. The setup consisted of magnetically tracked props in form of a *doll's head* for head viewing, a *cross-sectioning prop* to adjust cutting planes and a *trajectory-selection prop* to indicate the desired approach to a surgical target in the virtual head (Figure 22). The presented 3D image of the patient's head on the monitor could be rotated by rotating the doll's head, held in the non-dominant hand. Using the dominant hand the surgeon can define either cutting planes for the visualization of the 3D data or determine trajectories for a target. In compliance with Guiard's observations the non-dominant hand fulfills the rough, low frequency task of positioning first, then the dominant hand accomplishes the fine-grained, high frequency tasks. A major result of this work was also the introduction of instrumented props (stand-ins) as a general metaphor for interaction devices that correspond in physical properties with the virtual counterpart for easier acceptance and improved performance.

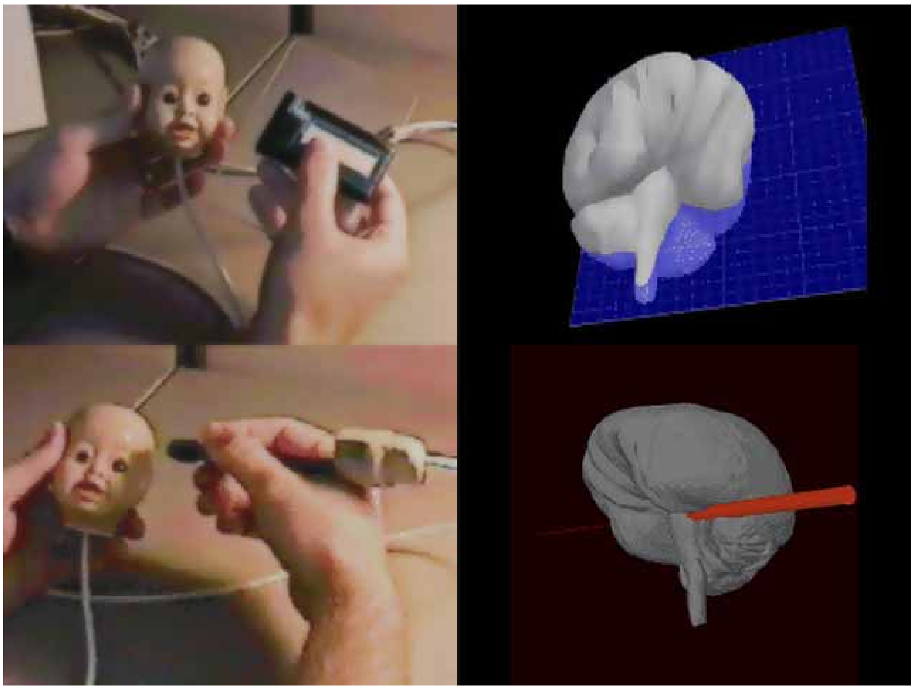


Figure 22. Neurosurgical planning with props. Using the doll's head and the plane a cutting plane can be defined (upper row). Using the trajectory tool a trajectory is set in the lower row.

Billinghurst presents in [Billinghurst, 1997] the *3D Palette*, a virtual-reality content creation tool. The physical interface is similar to the *3-Draw* interface, however the palette is a pressure sensitive Wacom tablet and the user wears CrystalEyes shutter glasses for stereoscopic output on a monitor. Commanding is supported by voice input for single spoken phrases.

Shaw describes in [Shaw, 1996] an interface using two modified magnetic tracking sensors. Contrary to the above presented approaches Shaw does not use props as interaction devices. In fact the presented *THRED* system uses two custom made 3-button Polhemus receivers, called *bats*. The user interacts using the bats in both hands switching between different contexts using the buttons. The direct manner of object manipulation improves user performance especially when performing complex 3D operations.

MultiGen's *SmartScene* product as published in [Mapes, 1995] uses two magnetically tracked data gloves instead of bats. Using the gloves the user can perform a variety of 3D operations and gestures. It concentrates on symmetric two-handed techniques for scaling, rotating, and stretching objects and navigating throughout the scene. Users can also align objects with both hands via anchors and constraints.

Cutler et. al. describe in [Cutler, 1997] a wide variety of different object manipulation tools that can be used with the *Responsive Workbench* setup. The basic building blocks for the interface are two Fakespace PINCH-gloves and a stylus, all tracked with a 6-DOF Polhemus sensor. User's were allowed to use one or two devices, whereas in case of two input devices both combinations of glove-glove and glove-stylus were encouraged. The categorization of the long list of tools is oriented on Guiard's classification of manual activities into unimanual, bimanual symmetric and bimanual asymmetric actions. In this work results not only reinforce Guiard's observations on asymmetric division of labor, but also noticed that users performed in comparison to a symmetric pair of gloves better when they used the asymmetric combination of glove and stylus.

A completely different approach to 3D-interaction is discussed in [Zeleznik, 1997]. Zeleznik et. al. present a metaphor using two essentially 2D input devices simultaneously for 3D object manipulation and camera control. Their conclusion show that a careful designed mapping of 3D parameters onto 2D devices can improve efficiency of complex manipulation tasks, whenever the metaphor used is common enough to users. Using entirely independent degrees of freedom on the two devices results in confusing set-ups and users feel disoriented. Again the observations of Guiard seem to hold, the authors report: "The best choice of mappings seem to be the ones that have the strongest physical analogs. Thus, techniques in which one hand pins down a point in the scene, and the other hand manipulates relative to that point seem to work particularly well."

2.4.3 Two-Handed Interaction in Immersive Environments

Desktop-based bimanual interfaces showed how asymmetric division of labor can improve performance, but also pointed out the fundamental importance of carefully designed interfaces. Immersive environments place even more requirements on interfaces. Participants of immersive virtual environments are visually cut off of the real surroundings. User interfaces for such scenarios have to consider that relocation of devices is hardly possible without a good virtual representation. Physical property of the hardware plays a significant role in the suspension of disbelief of the presented virtual environment.

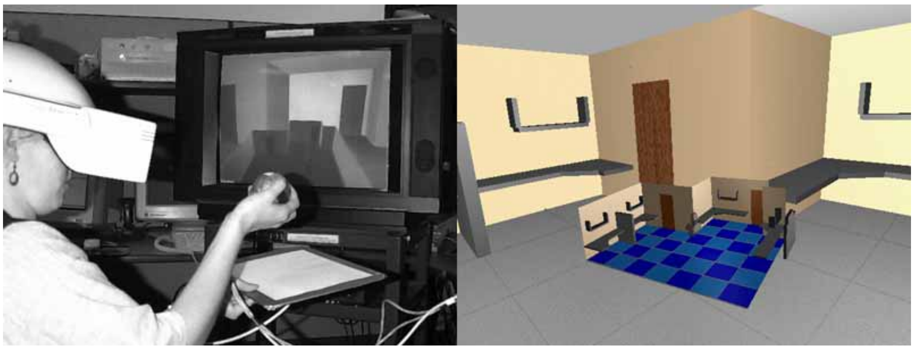


Figure 23. Virtual Reality on the *WIM*. Left image shows the actual interface. Right image is presented to the user in HMD.

Sowizral and Angus show in [Sowizral, 1994] and [Angus, 1995] how to address this problem in an immersive setup at the Boeing VR group for a maintenance task. They use a 3-button mouse, equipped with one Polhemus Fastrak receiver for interacting in 3D. This *Virtual Tricorder* is the handle of a paddle-like interface, where a square display surface is augmented in addition to the real prop. Thus a device was created that gives sufficient tactile feedback for holding the device and is flexible enough to fulfil different interaction tasks. Applications are controlled either with the buttons on the handle or with the other hand. The index finger on the user's other hand, or a stylus held in that hand, is also position tracked. The application encounters intersection of the virtual display and the pointing device, so 2D application-control can be performed from within the virtual environment. However, the drawback of this approach is that the virtual display surface gives no haptic feedback of the performed operation.

A different problem is addressed by Pausch et. al. [Pausch, 1995], [Stoakley, 1995]. Navigation in immersed virtual environments may become a difficult task for some users, since adaptation to new metaphors requires an introductory phase. In addition to the first person view the developer of the *World in Miniature* (WIM) metaphor supply a *God's eye view* of the life-sized surrounding space on a hand-held clipboard (Figure 23). Navigation, locomotion and object manipulation can be achieved at different scales by directly manipulating objects on the WIM with a button-ball held in the user's other hand and getting feedback on the scale of immersion. Edwards and Hand [Edwards, 1997] describe similar approaches in their work about the prototype of their user interface *MaPS* for navigation planning and viewpoint manipulation, which they implemented as an extension to immersive VRML2 [Hartman, 1996] browsers. In their implementation the user interface is part of the virtual environment itself and supports a wide variety of navigation metaphors. The employed *VirtualFlexor* interface device is very similar to the Virtual Tricorder approach in [Sowizral, 1994], [Angus, 1995], and [Wloka, 1995], it also extends the real device with virtual maps. A more recent work of Poupyrev presents in [Poupyrev, 1998] the *Virtual Notepad*, a two-handed combination of a small pressure sensitive tablet and a data-glove (Figure 24). The work focuses on taking handwritten notes in an immersive environment for annotation. Application control is achieved using one-letter commands written directly on the surface of the panel.

Figure 24. *Virtual Notepad* – allows taking notes in an immersive environment.

2.4.4 Summary of AR Systems and Interaction Metaphors

As result of this survey on augmented reality systems and two-handed interaction techniques a summary sheet could be created as shown in Table 2. Sometimes it is hard to distinguish between a whole system and a standalone interaction metaphor. Because of this, the name in the first column of the sheet refers to systems and/or to interaction metaphors. In one case (Virtual Tricorder) there is also a double naming in literature. When a metaphor is two-handed, the mark for the one-handed interaction type was only set if authors explicitly stated this. The categorization of interaction tasks performed with the device is divided into object manipulation, viewpoint manipulation, and system control according to [Hand, 1997]. However in some cases – when the device is used in different applications – affiliation to one or the other group is application dependent. In this case the table tries to cover all possibilities of the device.

Summary of Presented Augmented Reality Systems and Bimanual Interaction Metaphors

System or Interaction Metaphor	Published in	System Type					Interaction device			Interaction type		Interaction tasks		
		Desktop VR	Projection	HMD - immersive	Optical ST-AR	Video ST-AR	2D	3D props	Glove	One-handed	Two-handed	Object manipulation	Viewpoint manipulation	System control
3D Palette	Billinghurst, 1997	•						•			•	•		•
3DM	Butterworth, 1992			•				•		•		•	•	•
3-Draw	Sachs, 1991	•						•			•	•		
ARC (1)	Webster, 1996			•				•		•				•
CAVE/Wand	Cruz-Neira, 1993a		•					•		•			•	
Chameleon (2)	Fritzmaurice, 1993					•		•		•		•	•	
HARP (3)	Lindemann, 1999			•				•			•			
KARMA (2)	Feiner, 1993				•		•							
MaPS/VirtualFlexor (4)	Edwards, 1997			•				•		•			•	
Netra (5)	Goble, 1995	•						•			•	•		
Personal Interaction Panel	Szalavári, 1997 Schmalstieg, 1998		•	•	•			•		•	•	•	•	•
Personal Workspace	Imagination, 1999	•					•	•			•	•	•	•
Responsive Workbench	Cutler, 1997		•					•	•	•	•	•		•
Shared Space (6)	Billinghurst, 1996			•	•							•		
T3	Kurtenbach, 19	•					•				•	•	•	•
THREAD (7)	Shaw, 1996	•						•			•	•	•	•
Touring Machine (8)	Feiner, 1997				•		•			•			•	•
TransVision (2)	Rekimoto, 1996					•		•		•		•	•	•
VAnno (9)	Harmon, 1996			•				•		•		•	•	•
Virtual Notepad (10)	Poupyrev, 1998			•				•	•		•	•		
Virtual Tricorder	Wloka, 1995			•				•		•		•		•
Virtual Tricorder/Paddle (11)	Angus, 1993 Sowizral, 1994			•				•		•	•		•	•
Virtual Workbench	Serra, 1995	•						•		•	•	•		•
WIM (12)	Stoakley, 1995			•				•			•	•	•	

(Additional info: 1 - architectural construction task; 2 - spatial manipulation of device causes viewpoint manipulation; 3 - experimental study of two-handed manipulation; 4 - variety of navigation metaphors in VRML2 scenes; 5 - object manipulation covers o. positioning and selecting cutting planes; 6 - documents are selected with hmd-fixed cross-hair cursor; voice commanding; 7 - 2 bats - Polhemus receiver equipped with 3 buttons; 8 - head movement causes viewpoint manipulation; 9 - object manipulation covers manip. of virtual annotations; 10 - performed task is handwriting; 11 - second interaction device is tracked hand; 12 - combination of buttonball and clipboard)

Table 2. Summary of presented Augmented Reality Systems and Bimanual Interaction Metaphors

2.4.5 Basic Design Guidelines for AR Interfaces

From the above review of different concepts it is possible to extract basic design guidelines for a new two-handed interface for augmented reality applications:

◊ *Simple everyday metaphor* – The underlying metaphor should be taken from real-world applications or everyday knowledge, because users can then use their skills from these applications.

◊ *Abstract from original ideas* – Metaphors should always be a derivation from original sources of ideas and not involve nested levels of abstraction.

◊ *Metaphor not overloading user* – The selected metaphor should not induce additional cognitive load and the interaction with the devices should not distract from the task to be performed.

◊ *Carefully chosen props* – Props should be a simple abstraction of the underlying real-world devices.

◊ *Correspondence of virtual and real representation* – The design of similar props in the real and virtual part of an augmentation helps to suspend disbelief in the presented mixed reality. Also the virtual representations should be well registered with physical devices to avoid conflicting sensual perceptions.

◊ *Two-handed interaction should harmonize with Guiard's observations* – Several researchers proved the generality of Guiard's observations on asymmetric division of labor in human skilled bimanual action, the bimanual frame of reference, and how a cautious design of a two-handed interface can improve user performance for certain tasks.

◊ *Consideration of ergonomic factors* – Fatigue and other kinds of ergonomic discomfort arising using interfaces can quickly lead to embarrassment and frustration, therefore ergonomic considerations like weight and size should influence design proposals.

The following chapters will show how concept and implementation of the Personal Interaction Panel was accomplished using this resipe.

R. v. d. Weyden – St. Luke paints the Madonna.
(~1450)

Chapter Three - Conception

3.1 *Fundamental Design Goal*

The previous chapter outlined current research and development in the field of *Augmented Reality*. This technology will play a significant role in the future of how we interact with the growing number of computers in our environment. In the next decades this *parallel information world* of connected computer networks will become integral part of our everyday life and will more influence our way of living, acting and thinking, than anything else before in history of mankind.

To stay in contact with this virtual part of our world we have to introduce new powerful paradigms for interaction and communication. AR technology together with the research on *Ubiquitous Computing* tries to give technological answers to the prophecies of this bright future and shapes this tomorrow's world. Current work contributes to this stream with the introduction of a new interface for augmented reality. The *fundamental design goal* of this work is to:

Fundamental Design Goal

> Create a general interface for interaction with virtual content in augmented reality setups, so that interaction is as natural as possible and users are supported in the best possible way to perform the tasks they intended to.

3.2 *The Personal Interaction Panel*

Investigating interaction tasks in AR systems, analyzing their advantages and drawbacks we outlined a concept for a new device. We introduce a two-handed augmented input device that consists of a real and a virtual part. The real part is composed of a notebook sized hand-held panel and a stylus that allows users to use everyday knowledge and subconscious skills about manipulation. Mankind uses this type of devices for eons of time, historical ancestors reach back as far as the ancient times of Egypt; accompany us in form of *slate* and crayon, palette and brush and are part of our modern life in form of clipboard and pen (Figure 25).

The virtual part completes the real part with three-dimensional augmented information displayed directly at the surface of the panel. To represent the suitability of our tool for a wide range of interaction styles and to express its personal character we called this hardware setup the *Personal Interaction Panel* (PIP).

Figure 25. Historical ancestors of the *Personal Interaction Panel*: Thot – Egyptian god of writing and science, *slate* and chalk, clipboard and pen.

Definition of the Personal Interaction Panel

> The Personal Interaction Panel is a general two-handed augmented input device that consists of a real and a virtual part, supporting a wide range of application scenarios. The real part is composed of a notebook sized hand-held panel and a stylus. The virtual part holds three-dimensional information at the surface of the panel that can be directly manipulated with the stylus.

Using this everyday metaphor no special abstraction is needed for the basic understanding of the interface. It unifies the advantages of simplicity and flexibility and thus supports a multitude of interaction styles. The physical nature of the pen and panel makes it a very simple, yet effective and precise device for interaction, that supports tactile feedback and has good ergonomics. However, the surface of the panel is a virtually unlimited interaction surface and an information display of computer generated (augmented) information (Figure 26).

There are many different possibilities to use the PIP as interaction tool in the augmented environment. In the following chapters we will show features for general object and viewpoint manipulation tasks, application specific and miscellaneous control tasks.

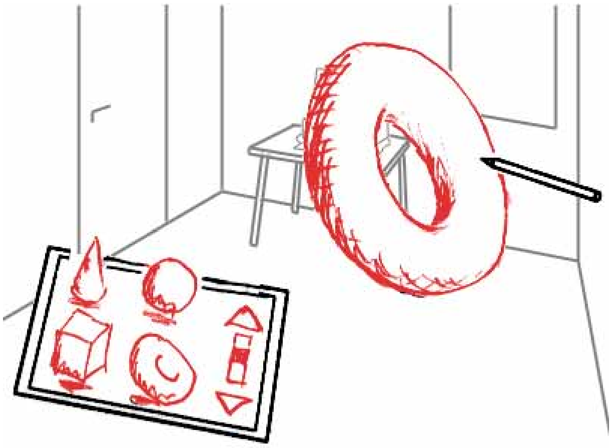


Figure 26. Concept of the *Personal Interaction Panel*. Everything drawn in red is augmented to reality, which is shown here in black.

3.3 PIP Variants in the Design Process

In the design process the conceptual definition of the Personal Interaction Panel was refined in several ways. Three actual design variants were theoretically elaborated. We took pros and cons of the variants into account considering ergonomic and suitability to our basic definition of the problem. All design variants use soft- and hardware technology that is definitely or is roughly available. According to the definition all three variants are composed of a lightweight, notebook-sized hand-held panel and a pen. Furthermore both panel and pen are tracked in position and orientation either by standard magnetic trackers or by optical tracking for the correct registration of real and virtual world.

3.3.1 Pressure-sensitive Flat-Panel Display with Pen Observed with Stereoscopic Glasses

The setup shown in Figure 27 resembles hardware as shown in [Mel, 1988] and [Weiser, 1991] and an enhanced version of today’s commercially available hand-held palmtop computers like Apple’s Newton Message Pad or the 3Com Palm VII. Using LCD shutter glasses or passive polarized glasses, the continuously updated computer imagery on the display shows three-dimensional images appearing to escape from the flat display, floating above it’s surface as if it would be a portable “Responsive Workbench” [Krüger, 1995]. Being portable the display offers exploration of an augmented environment as proposed in [Amselem, 1995] and [Rekimoto, 1995]. The pressure-sensitive surface allows not only click-actions, but also fine-grained sensing of pen actions, which can improve the interaction capabilities. The disadvantage of this technology is besides the demanding hardware, the limitation by a relatively steep viewing angle, due to the display technology used. With a significant improvement in size, availability and cost of the recently presented autostereoscopic displays [D4D, 1999] this variant has also the potential to present 3D images without any glasses.

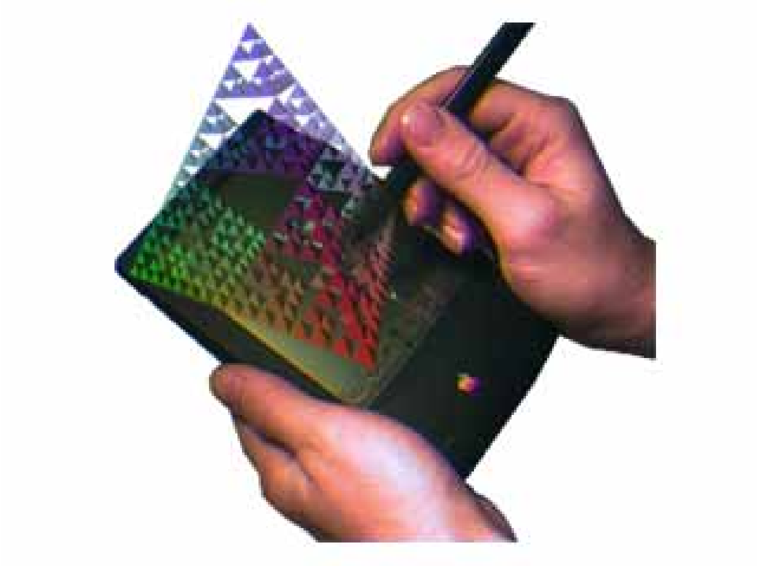


Figure 27. Pressure-sensitive flat-panel display with pen observed with stereoscopic glasses.

3.3.2 Pressure-sensitive Flat Panel with Pen and a See-through Head Mounted Display

All benefits from using a pressure-sensitive device remain in the setup shown in Figure 28, only the display surface moved from the surface of the panel to the eyes. Synthetic imagery is displayed at real time to the user in a see-trough head-mounted display (HMD). The level of augmentation and immersion increases, since in a multi-user scenario users without HMDs do not see anything on the board, yet they recognize the panel being an input device supporting computer-human interaction. Furthermore a HMD system allows not only projection on a panel but extends augmentation into the whole environment. A drawback of this system is, that full interaction is limited to a specific part of the panel. Despite the benefit of precise interaction on the panels surface the fatiguing weight of even small panels, like the Wacom ArtPad II can constrain work and may be disturbing for the user over a longer time of usage.

Figure 28. Pressure-sensitive flat-panel with pen and a see-through head mounted display.

3.3.3 "Dumb" Panel and Pen in Combination with a See-through Head Mounted Display

In this variant neither panel, nor pen has any built-in hardware intelligence apart from the trackers mounted on them, similar to all other versions above (Figure 29). The three-dimensional imagery is presented to the user on a see-through HMD, in accordance to his actual viewpoint and viewing direction. Position and orientation tracking of all three parts (panel, pen and HMD) allows the correct evaluation of spatial relations for perspective matching of the real and augmented environment. The physical properties of the devices support exclusively tactile feedback to the user, enriching the interaction.

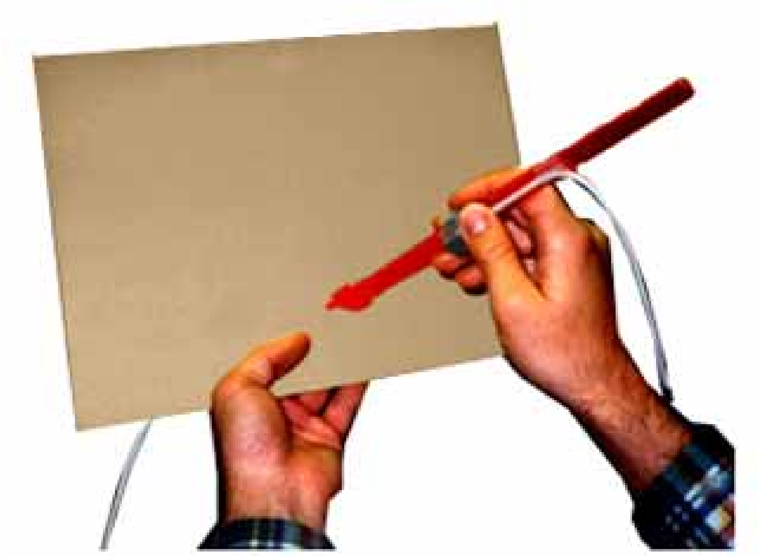


Figure 29. “Dumb” panel and pen in combination with a see-through head mounted display.

Although being technically plausible the first variant requires still extensive hardware development, which would implement our idea hardwired. Furthermore we see the viewing angle constraint of current LCD technology to be very hindering, however this might be resolved in future. Whereas a simple selection signal for a pick event on the panel, coming directly from the hardware would be helpful, the second variant has several limitations. The sensitive area is fixed in its size and properties. Furthermore most pressure sensitive panels are too massive for continuous hand-held usage and the device interferes with current state of the art magnetic tracking systems.

For our implementation we used the third variant, offering the widest spectrum of degrees of freedom in designing the interface itself and being the most flexible for rapid prototyping, as it has hardly any hardware limitations and software design determines the full functionality.

3.4 *Features of the Physical Interface*

3.4.1 "2D in 3D"

Our combination of an interface with an empty flat surface and sophisticated see-through HMD display technology allows the projection of three-dimensional imagery everywhere around the user, but gives also an opportunity to recycle 2D GUI elements. The *Personal Interaction Panel* supports this mixture of the 2D desktop metaphor and the 3D display, so 2D interaction and three-dimensional direct manipulation are done in parallel. Unlike many other interfaces it implements a 2D interface *in* 3D, like a notebook or a piece of paper with its flat surface in the real world, rather than a combination of 2D *and* 3D, requiring a mental switch from flat to spatial. In our everyday life we are used to work "on" 2D surfaces like paper, desk and blackboard. We also organize interaction elements on 2D surfaces – like for example knobs and buttons on a switchboard or a keyboard. This schematic organization of control seems operators to help to keep a good overview when using these interfaces. In addition the same flat concept led to the conventional 2D computer desktop interfaces.

In more detail this concept of 2D *in* 3D implies that a conventional 2D computer display can be projected onto the board, supporting the 2D desktop metaphor better than "flying menus" or buttons [Butterworth, 1992], [Harmon, 1996], [Jacoby, 1993]. Having the tactile feedback of the surface and the fine-grained interaction of one hand in the frame of action of the other hand, users can easily interact with desktop elements. Note the interface does not float around in space, because it is still "directly" connected to the user's hand. This imitates holding it in the hand. Thus locating user interface elements becomes rather easy. Dialog boxes for the manipulation of parameters are laid onto the panel and selection or generation of other events is done with the pen. In addition to all these 2D user interface elements 3D user interface elements such as 3D widgets are incorporated in the interface of the PIP, supporting three dimensional tasks better than 2D elements. Rising the tip of the pen from the surface of the panel makes it to a six-degree of freedom manipulation device for direct manipulation or selection. In combination with the panel or without it, similar to real world pens, we use for pointing or as an aid to envision other objects in discussions and the 'reach in with a hand-held stylus' metaphor described in [Serra, 1995]. Again, the 2D *in* 3D axiom makes this mixture natural and no cognitive switch is necessary.

3.4.2 Asymmetric Bimanual Interaction

Looking at the basic design guidelines in chapter 2.4.5 the PIP interface conforms previous results in bimanual action [Guiard, 1987], [Hinckley, 1994b], [Hinckley, 1997b], [Kabbash, 1994] and [Sachs, 1991]. The panel defines a base in three-dimensional space with the non-dominant hand and determines the relative frame of action of the dominant hand. This important feature of the interface promises better performance results already in the design process. Previous tests showed that target acquisition accuracy and time in this relative frame is superior compared to absolute frames of reference for interaction. The ability to act in a body related frame of reference – called *proprioception* – is very helpful for navigation in an environment, but results in lower performance when used for direct object manipulation in tree-dimensional space.

Apart from the below proposed interaction metaphors the asymmetric design of the interface itself (different devices in the dominant and the non-dominant hand) induces an asymmetric division of labor. This subconscious everyday skill causes that users first position the pad and do the finer interaction after that with their dominant hand, as the cognitive load of coarse and fine tasks is asymmetric. Our interface is well suited for both right- and left-handed users; a system design issue was, not to incorporate preferences.

The interaction with the Personal Interface Panel profits from the existing *kinaesthetic feedback* that allows us to know the position of our limbs relative to our body. This is a very important feature for relocating the devices during interaction since the user does not have to hold the devices in front of him. A typical interaction scenario would be manipulating some controls on the panel, releasing the device and then looking at the augmented environment to get feedback of the changes. Now users can again lift the devices into the field-of-vision of the HMDs or tilt the head with a pitch motion to see the devices. Without *kinaesthetic feedback* would have to look around to find the devices again.

3.4.3 Haptics

Our pair of magnetic tracked pen and pad gives sufficient *tactile feedback* and is familiar to inexperienced individuals, enabling them concentration on the tasks to be performed. Rather than offering virtual devices for manipulation tasks we propose *extended devices*. Extending the real world tools by added virtual shape and functionality while preserving parts of the tool being always identical, users feel to hold all tools in their hands. Our implementation with see-through head-mounted displays augments synthesized imagery onto real world objects, but – as illustrated in later sections – this interaction technique serves well in both virtual and augmented environments. Tactile properties of the interface device are kept in a virtual environment, so that coherence in shape of the real tool and the virtual tool helps satisfying the need for convincing stimuli like also shown in [Hinckley, 1994b] and [Wloka, 1995].

Coincidence between real props and virtual representation of the tools made it natural to lay down and find again the tools, even in the environment of our crowded lab. Despite our implementations gather around augmented reality applications, most of the Personal Interaction Panels functionality can be translated to virtual environments. Since the "dumb" panel does not require to be seen when represented accordingly in the virtual space, full immersion of the user into a virtual environment is also possible.

Size correspondence of real and virtual interface is important in the case of the panel. We varied the pen thickness in different applications, which did not make an impact on the sense of correspondence.

The surface properties of the actual setups as shown in chapter 4.1 were different. Users expressed to like a little roughness of the panel, so sliding actions on the surface would provide the tactile feedback of scratching above the surface.

Hardly any test person did report problems of fatigue. They immediately realized that wearing a HMD allows to change the point of view rather quickly, so lowering the arm, using the PIP sitting or even placing it for a short time on a desk does not have an impact on its use.

3.5 *Metaphors for Interaction Tasks*

The discussion of the physical interface lead to a theoretical setup for the system hardware. However, the virtual part of the Personal Interaction Panel is entirely depends on the underlying software components. To complement the interface a wide variety of interaction metaphors were investigated that could benefit from the hardware setup. The right combination of the interaction metaphors with the proposed hardware should create a unique tool that combines physical and virtual attributes and meets the basic design goals governing this research work.

One basic issue we will follow is to avoid context switching from the virtual part of the augmentation to real world. For this reason the virtual overlay on the PIP has to behave like the real world. In this sense the fewer new metaphors and abstraction levels we introduce the more natural the interaction will be.

As the PIP is a multi-purpose interaction device, the wide range of applications has to be categorized for further investigation. We classified user interaction tasks using Hand's categories presented in [Hand, 1997]. The presented interaction metaphors are categorized in *object manipulation*, *viewpoint manipulation* and *application control* respectively. In the beginning we will follow this classification and show the usability of the PIP to certain general tasks. Later we describe metaphors for special tasks and specify certain features in more detail that govern the subsequent implementation.

3.5.1 Object Manipulation Tasks

Modeling of objects has been an issue from the beginning of computer-human interaction. Working with objects directly in three dimensions rather than with 2D projections improves understanding of shape and relations. In augmented environments one has the possibility to compare a real model with the modeled copy of it. Overlaying of real and virtual model or extending a real object with virtual parts (e.g., seeing ultrasound imagery in the patient [Bajura, 1992]) or annotating real world objects [Rose, 1995] employ the real capabilities of AR.

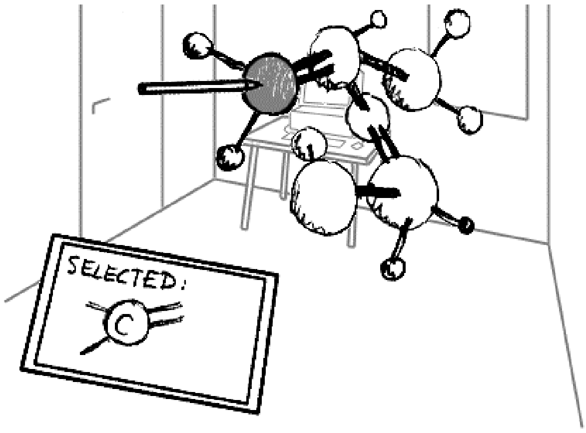


Figure 30. Direct selection of objects by inserting the pen into the “floating” model (background illustrates augmentation).

Basic object manipulation tasks, like *object selection, transformation* and on a higher level *drag and drop* of objects in three dimensions have been investigated by many research groups [Butterworth, 1992], [Conner, 1992]. In our setup, the pen alone is used for 3D pointing-, selection operations or direct manipulation of the displayed model, where a 6D mouse is normally used. This feature is seamlessly integrated within the extended PIP functionality, so that the PIP supports a superset of “standard” 3D operations in virtual and augmented reality.

Furthermore *showing the selected item simultaneously* on the panel (Figure 30) enriches selection of objects floating in the augmented scene. The contemporary display permits adding additional information to the selection (e.g., physical properties like volume, etc.) on the panel.

The PIP is capable to be used as a visible *3D clipboard* carrying a collection of 3D or 2D data items that are shown above the panel’s surface and can freely be accessed by the user. Objects may be dragged out from the surface of the PIP and directly placed or moved in the augmentation (Figure 31). This mechanism gives a natural interactive feeling of handling spatial aligned data. Once more the tactile feedback supports interaction, giving the user the feeling, to hold the items in the hand. Correct placement of the objects on the surface, allows to picking on the panel, as if force feedback would have been added.

Besides direct manipulation, handling objects with abstract metaphors like *3D widgets* can be useful for certain applications. Unlike many other works, the manipulation draggers are in our case not attached to the object but to the surface of the PIP, enabled by the mixture of 2D and 3D desktop elements, described above. Advantages of two-handed interaction and in particular the PIP, like frame of reference and panel attached interface elements, improve work with these abstract tools Figure 32 and Figure 33.

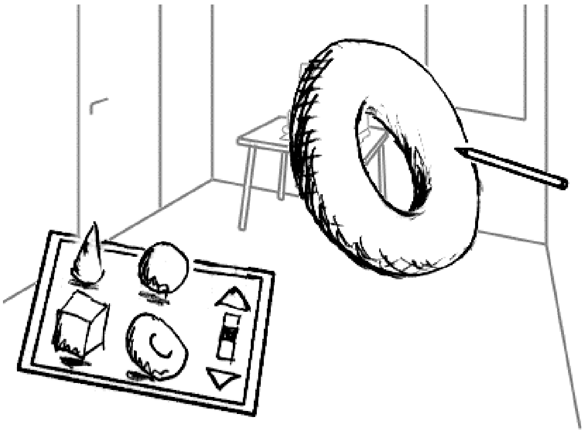


Figure 31. “Drag & Drop” objects from a clipboard into 3D space.

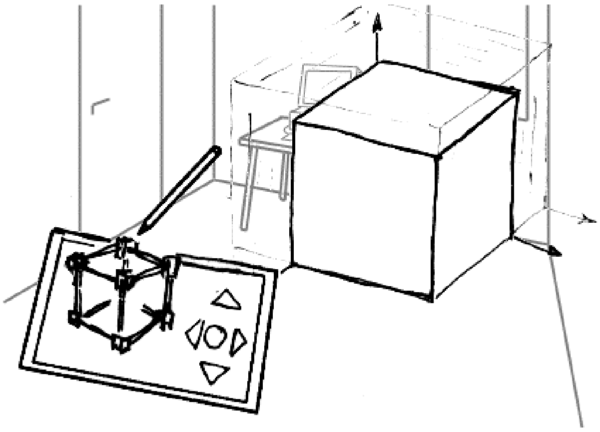


Figure 32. In addition to direct manipulation, widgets can be used for exact scaling …

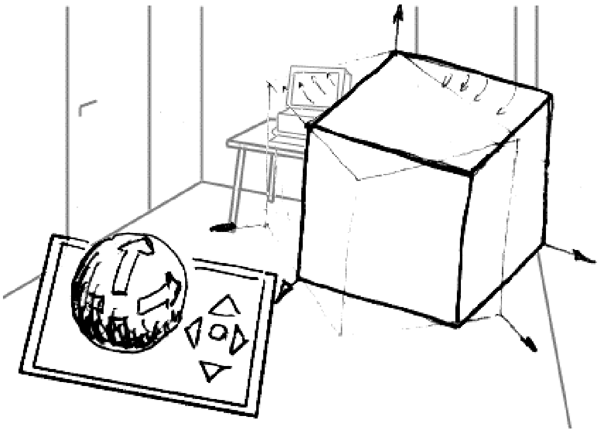


Figure 33. … or rotation of objects.

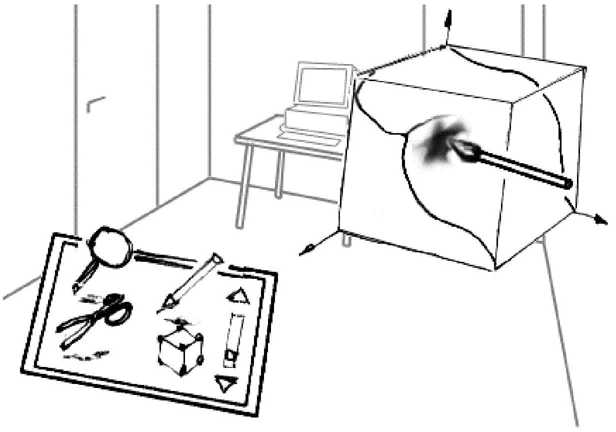


Figure 34. General PIP tools (coloring brush).

A *3D tool-palette* in Figure 34, comparable to the object browser clipboard in Figure 31, contains all basic functionality of the PIP. Among others features like cut and paste tools, *Magic Lenses* [Bier, 1993] and coloring brushes are supported. Attaching these augmented tools to the tip of the pen like shown in [Serra, 1995], the user is given the feeling of holding an "extended" tool in his or her hand.

3.5.2 Navigation Tasks

As object manipulation concentrates on handling and editing of objects, navigation is necessary for *changing viewpoint position and orientation* in order to explore a specific part of the environment. Hand generalizes in [Hand, 1997] the term of navigation by saying: *"Viewpoint manipulation encompasses the tasks of navigating in the virtual environment by movement of the point-of-view (i.e. the camera or 'virtual eye'), as well as controlling the viewing parameters such as Zoom Factor and Field of View."*

In our augmented reality setup the user is wearing a HMD, 6DOF-tracked with magnetic sensors. Movement of the head corresponds automatically to a change of the viewpoint, thus coincidence between real and virtual imagery is ensured.

Additional to the own viewpoint movement, navigation metaphors described in [Hinckley, 1994a] are supported in our AR setup with the PIP interface. The "eyeball-in-hand" metaphor is in our case a "*look where you point*" viewpoint, for that virtual camera position and orientation is defined with the pen, while the off-screen rendered camera image is displayed on the panel, as shown in Figure 35.

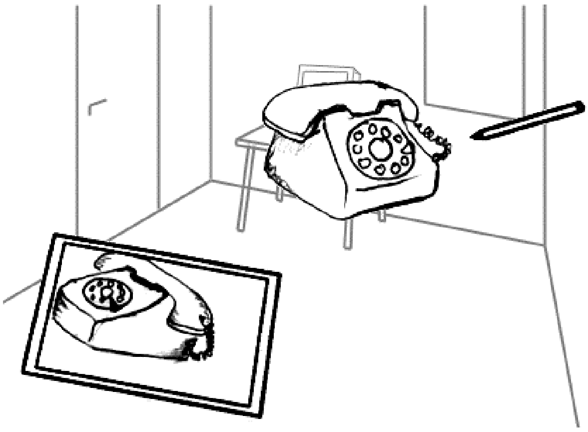


Figure 35. Camera positioning with the pen.

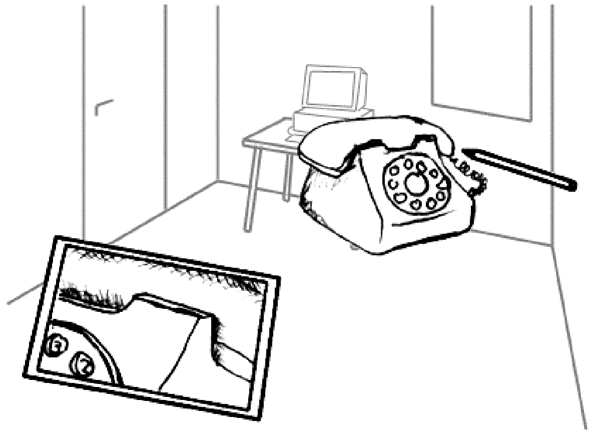


Figure 36. Enlarged view of a detail.

Magnified details, even from the inside of a simulated object, can enhance working in the augmented environment and are supported by the PIP. Figure 36 shows a zoomed detail on the PIP, while observation of the whole object is still possible in parallel.

During navigation the panel can act as a *virtual photo camera* to take a "snapshot" of the scene's state. The currently displayed image or detail is immediately frozen on the panel's surface and e.g., printed out on a remote printer.

Considering that AR applications need the correspondence between real and virtual environment, navigation and locomotion independent from the real viewpoint is not used, as it would destroy the augmentation. However, for specific VR applications the PIP is able to support further navigation metaphors like specifying direction of movement with the pen or *"spaceship" control gadgets* (2D buttons or 3D widgets) on the panel's surface.

Map-based techniques complete the list of possible navigation methods. The "scene-in-hand" metaphor similar to the WIM interface [Pausch, 1995] and [Stoakley, 1995] and the MaPS interface in [Edwards, 1997] can be supported, showing an overview of the surrounding environment. The displayed *map can be scrolled with arrows* on the panel's surface or grabbed with a pen-click and dragged it to any direction causing the map to slide (*grab & drag*). The representation of the user in the environment is fixed at a location (e.g. center) of the panel. As movement is translated in an absolute way, releasing and grabbing again could be necessary for larger distances. To overcome this drawback an *avatar grab* method is introduced similar to the presented approach in [Pausch, 1995]. The avatar representation of the user in the VE on the panel is grabbed and "drawn" through environment relative to the grabbing point. While manipulating the avatar the map is scrolled accordingly into the other direction so that the representation never "falls off" the map. The speed of this scroll can vary similar to the *depth modulated flying* approach in [Ware, 1997]. However speed magnitude would be controlled with the distance of the avatar from the grabbing point. Moving the panel towards or away from the eye during navigation causes zooming of the map for different speeds of locomotion.

It is very important to summarize that all navigation metaphors presented here in combination with the Personal Interaction Panel have one in common. While using whatever metaphor for navigation with the PIP, the user still keeps the connection to the environment and does not have to make a cognitive switch, because the own view is kept and the navigation display is on the panel.

3.5.3 Application Control Tasks

The design of overall controls is crucial for a system. The user may not be forced to need very special skills for the general controls of the application, as he wants to concentrate on the topic of the application. Immersive augmented applications need easy to understand controls, which have to be "inside" the augmentation. For the feeling of good immersion it is very important not to leave and join the augmentation for operations like reconfiguration of the system, starting a new session, etc. Many solutions transfer conventional application controls to AR or VR systems, like flying menus [Butterworth, 1992], [Harmon, 1996] which have drawbacks, not considering the three dimensional behavior of this applications, the reduced absolute target acquisition skills in space and the lack of tactile feedback [Hinckley, 1994a]. Other examples attach menus to the interface [Wloka, 1995], so target acquisition has only to be relative, enhancing performance.

The PIP offers the possibility to contain and manipulate all the necessary controls in a desktop manner as described above and shown in Figure 34. General controls for an AR system like *Studierstube* can easily be made available by the PIP, so reconfiguration of the application can largely be achieved without leaving the augmented environment. A tool-palette Figure 37 groups functions and make them easily accessible. A conventional 2D computer display can be projected onto the board similar to [Angus, 1995], supporting a 2D desktop metaphor better than "flying menus" so traditional 2D user interaction and parameter manipulation is possible. In addition to "flat" 2D user interface elements, three-dimensional widgets that "float" above the panel's surface are supported (e.g., selection of a point on a sphere), clipboard functionality and drag-and-drop in 3D can also be implemented as pointed out above in the object manipulation section.

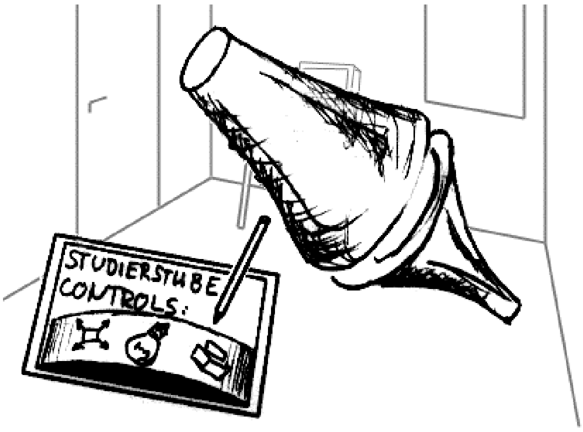


Figure 37. Controlling general parameters of an application, such as system setup, etc.

3.5.4 Precision Enhancing Techniques

Although being in fact an implementation issue we have to consider already in the conception phase the low resolution of HMDs and the lack of haptic feedback. These drawbacks decrease overall precision of systems based on this hardware. Additionally magnetic trackers provide noisy position and orientation measurements decreasing performance in object manipulation tasks. Due to that objects can not be placed very exactly in space. Especially, moving one object face-aligned onto another, which is a very often-performed task, is really hard to achieve. To *overcome the limited precision* of the applied VR hardware we enhance manipulation precision on the scene level, instead of applying quality enhancing techniques on raw tracker measurements, like filtering or prediction as done traditionally. We identified *snapping mechanisms* as a very powerful tool for aligning objects precisely, speeding up manipulation tasks.

Related work

Several solutions have been proposed for the problem of precise object placement. Collision detection [Cohen, 1995], an often used approach in common VR-systems, does not add very much value to that problem. The avoidance of interpenetration does only help a little for the alignment task and is computationally expensive.

In [Mine, 1997] *proprioception* is exploited to enhance the precision mainly of remote manipulation of objects which are out of reach of the user.

Bier's famous work on snap dragging in 3D [Bier, 1990] was one of the first solutions for direct 3D manipulation with high precision. Although his method is very intuitive, it can not be straightly used for direct manipulation, too many additional commands have to be set up during the task.

The whole field of constraint based modeling (summarized in [Juster, 1992]) deals also with high precision. There, constraints are used to automatically keep some kinematic relationships between parts of the scene. Some systems maintain a constraint graph to store this constraints for further use [Fa, 1993]. They describe a method to introduce new constraints into the graph during interactive manipulation.

Our approach

We use *face-snapping*, where objects, which are close to each other are aligned automatically by calculating an alignment constraint between the faces – the objects snap onto each other. If an object has to be placed onto another object, the user simply moves it close to the other and face snapping aligns them automatically.

Our method is similar to the interactive constraint based modeling technique proposed by Fa et. al. [Fa, 1993]. The objective of their work is to build up a constraint graph during object manipulation. A constraint recognition task finds possible constraints during the movement of one object and displays them to the user. If the user does not move further for a certain amount of time, the constraint will be set up in the constraint graph and maintained for further use.

In comparison to Fa's automatic constraint recognition process, our approach does not introduce explicit constraints, which can be inserted and deleted from a data structure. During dragging of an object by the users' pen for each frame all possible snapping conditions are checked and the one with the highest priority is performed immediately. The dragged object is moved according to the geometric constraints defined by the two geometries. Because the detection and calculation process is invoked after every pen movement, the two faces keep snapped until the snapping condition is no longer valid. If the user releases the object during a snap, the object stays aligned with the other, but no constraint is kept for the future.

Besides the described advantage of fast and precise object placement the snapping movement itself gives feedback for the user that he has placed the object correctly. So there is no additional effort needed to show that a positioning action has been completed. It allows different types of constrained motion by leaving one ore more degrees-of-freedom untouched after applying the snapping constraint.

3.5.5 Spatially Controlled Semantics

A special way of system control we have designed is the concept of *spatially controlled semantics*. A lot of actions while manipulating virtual content in augmented environments do have additional to their geometric also semantic meaning. This can be also observed in our real environment. For example putting a document in a file associates it to a group of other documents on a higher level, than just the spatial relationship. For example object placement using snapping is not purely a geometric alignment task. This type of manipulations also effects semantic meaning. The problem is to read the semantic meaning out of the geometric action. Mine et. al. [Mine, 1997] introduced the idea of gestures to identify this type of actions and trigger semantic meaning. One nice example was the interpretation of the movement of throwing an object over the shoulder as deleting this object.

Bukowski et. al describe in [Bukowski, 1995] a software framework to populate 3D environments with objects on the "WALKEDIT" desktop based 3D design system. They map 2D cursor positions into 3D space and enhance object placement with pseudo-physical behavior. In a second step they associate objects implicitly based on geometric characteristics, like distance to nearby entities. Associated groups are dynamic and can have hierarchical structure. This work has shown that "magic" – i.e. pseudo-physical rules – can enhance interaction with 2D interfaces and increase productivity.

Our approach

We introduce the general concept of *regions* as an extension to previous approaches. Regions are dynamic, logical groups of objects in a scene. They act as a container to hold groups together, identifying some kind of association of members. Regions can also be placed into regions, allowing the hierarchical association of objects contained in these groups. Regions are unambiguous to objects located in them.

To employ this organizing method in an augmented interaction scenario, regions are assigned to geometric objects with some spatial extent. Thus recognizing geometric conditions between this extent and arbitrary objects in the scene can form logical groups. An object is moved from one region into another if it is moved in 3-space out of the area of one region into the area of another region. There are three problems with this approach:

◊ It is very time consuming to calculate geometric intersections between regions and individual objects, especially if there are many regions in the environment.

◊ As regions not necessarily have a geometric extent associated and are only logical groupings, the user needs a separate feedback for entering and leaving a region.

◊ If regions overlap, there has to be a simple decision mechanism to decide for each object to which region it belongs.

For simple tasks like shown in our direct manipulation scenario of a game described in chapter 5.5.5, we identified the geometric condition to be identical with the snapping condition of two faces or a face and a plane.

Using this approach, *snapping is a mechanism to read out semantic meanings from the geometric actions*. The same interaction event can be used for precise direct manipulation and semantic control. Snapping constraints give the necessary visual feedback of docking. In parallel to the geometric docking a semantic action is triggered to associate the manipulated object to the target, where it was moved. We call this process of association to different logical groups *region transition*. Overlapping geometric conditions are resolved by snapping priorities, assigned knowledge-based. This is done in advance, in the design process of applications.

The theoretical design of the method for spatial controlled semantics read out from geometric actions should be a powerful technique for manipulation of objects in augmented reality environments, if objects represent not only geometry, but also are functional part of a simulation.

3.5.6 Privacy and Publicity

Most multi-user VR and AR applications present the synthetic environment to each user in the same way, albeit from different viewpoint or resolution due to LOD selection algorithms. Like early 2D collaborative systems they simply replicate the common database and show the same visual content. We intend to use the PIP interface in multi-user situations, where it is crucial that the Personal Interaction Panel becomes really personal. Multiple users collaborating in one augmented environment may communicate their ideas and share information. As described below in chapter 5.4.1, we are currently working on the *Studierstube* system to realize such a system for augmented scientific visualization. Even more than scientific visualization gaming in an augmented environment needs a privacy and publicity management of visual information. Induced by these requirements together with Eckstein and Gervautz we developed a security management approach, which is highly flexible, and suits different AR or VR application scenarios [Szalavári, 1998].

Related work

Smith and Mariani describe in [Smith, 1997] a mechanism to present subjective views in an existing distributed multi-user environment. Query results in the shared database are presented subjectively to users by assigning object modifiers to found entities. Thus relationships and representation of the requested data can be tailored to user specific needs and additional information is not cluttering up the scene for other participants.

Agrawala et. al. present the two-user Workbench in [Agrawala, 1997], introducing the potential to display customized views to two users. They also propose different partitioning techniques to present information.

While these solutions provide security for visual information, multi-user interaction in shared environments leads to even more complex problems. Broll offers in [Broll, 1995] a good overview on what distributed application might implement to handle concurrent object access. The paper identifies locking and the use of master entities as primary solutions for multi-user interaction in VR systems.

The aspect of allowing users to keep individual information on their PIPs implicates the sharing or concealing of this information, while the overall appearance of the augmentation should stay untouched. One concept to build a bridge between private and public is to place "public projection walls" into the environment. These walls are static, virtual objects in the augmented environment and are logically connected to one user's PIP, reflecting all changes immediately to the public. Thereby, a single user interaction tool can be transformed into a multi-user presentation spot, where personal ideas can be shared.

Another way to share ideas is to allow "giving" the content of one's PIP to other users. This can be achieved with the extension of the presented drag and drop mechanism to move the whole contents of one PIP may be logically moved to another users interaction panel.

The nature of our implementation environment implicates that all augmented information is automatically hidden for not immersed users. Participants in the immersed environment may see public objects standing around them, but private information on another user's PIP should be possibly hidden away. As every user wears a see-through HMD on his own and thus imagery is rendered for every user individually, filtering of parts of the scene, e.g. information on other users PIP, is easily achieved.

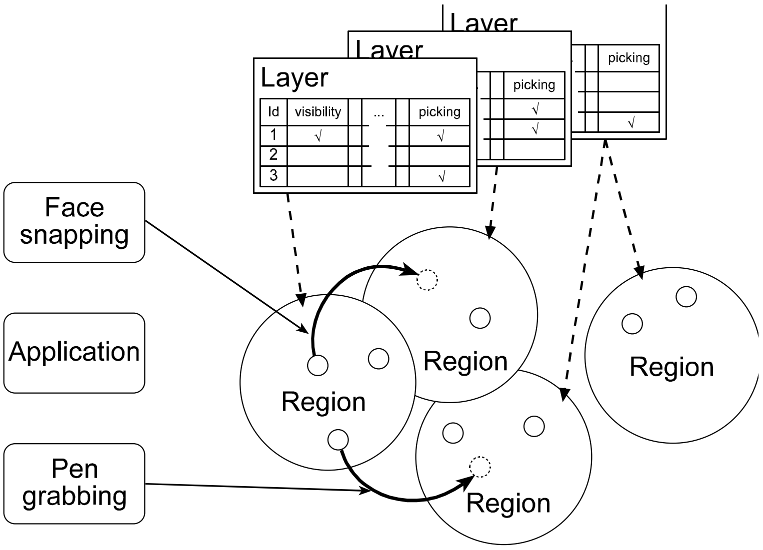


Figure 38. Object transitions between regions can be triggered by events coming from snapping, the pen or the application itself. Layers encode privacy information for players and are assigned to regions. Transition of an object (dots in the regions) into regions with different privacy, will change the security state of the objects automatically.

Our approach

For the security management we introduce a concept similar to that found in CAD or *GIS* packages for *layer*ing information. Our investigations lead to the result, that in most scenarios *groups of objects with the same security status* can be identified. So usually the number of different security statuses is much less then the number of objects, which have to be considered for security management.

Furthermore, drafting the system design for a gaming application described later we discovered that especially in gaming scenarios *security "presets"* can be found for certain parts of the game. These parts do not change during the play, e.g. the table is always visible for each participant, but one player's game-pieces remain only accessible for him- or herself. This defines security characteristics for logical parts of the game and thus for logical groups of game-pieces. We identified *regions* – described in the previous section – as the groups to hold a specific security state or preset for all objects in the logical group. This approach is simple yet powerful enough to meet most requirements. It is transparent to the application and does not cause additional management load as presented in Figure 38.

Private Help

Additional to simple modification of object appearance, the presented security layering mechanism is easily applied to achieve *private help*. Private help is any kind of help information or annotation similar to [Harmon, 1996], which can only be seen by the user, who requested the help. Other users should not see the help information and would be probably disturbed by it. Sometimes it is desirable that the application gives a hint to a specific user, which other users should not see, e.g. in a teacher – student collaboration, where only the teacher can see the right answer.

To implement private help all the geometry, which form the help information, has to lie on a special help-layer. Only the user, who should see the help information, has rights on the help-layer. Other users do not have rights on this layer and therefore they can not see, nor manipulate the help information.

The application domain of multi-player games is a good test-bed for our security management approach. For that reason we have implemented a multi-user game *Mah-Jongg*, which is described in chapter 5.5.5. Depending on the definition of layers, independent subjective versions of the same scene can be presented to participants providing a private view. The private space of one user is protected from other users, while public spaces allow access to everybody. Independent from visual appearance, manipulation or snapping characteristics may be hidden from others. Assignment of rights remains an open question and application programmers have to adjust security presets carefully to provide meaningful combinations. In direct manipulation scenarios like a game it is crucial to bring in ideas for security management possibilities, so that the interaction metaphor controlling this application gets accepted in multi-user environments.

3.5.7 Application Specific Metaphors

Scientific Visualization

As our first major application for *Studierstube* we developed a shared multi-user augmented research environment for scientific visualization, where we intended to make extensive use of the Personal Interaction Panel. Gröller, Wegenkittl in [Gröller, 1996] and Löffelmann in [Löffelmann, 1996] work with Fuhrmann [Fuhrmann, 1997] in our lab on the visualization and investigation of non-linear dynamic systems.

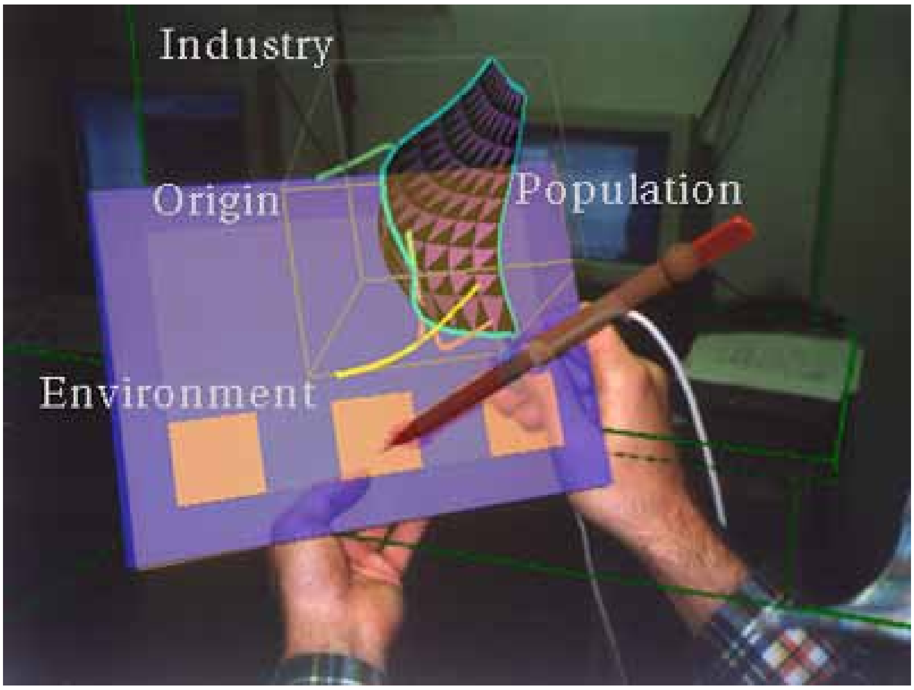


Figure 39. The "Wonderland" model on the PIP [Gröller, 1996].

In a cooperation we work on the visualization of stream surfaces, trajectories and local behavior of these systems in augmented environments. Figure 39 shows the model of the *Wonderland* econometric dynamic system on the PIP while performing interaction with augmented buttons projected on the panel's surface.

In this application the PIP can be freely configured with special interaction metaphors for scientific visualization. In the simplest case steering parameters of the underlying simulation can be mapped easily onto the PIP and modified with the pen.

Other metaphors include defining 2D cross sections and 2D projections with the panel as shown in Figure 40. Representations can be displayed right on top of the PIP and exchanged among researchers. The augmented reality setup also allows the use of an additional high-resolution CRT monitor for the display of high quality 2D images (e.g., the mentioned cross sections) without leaving the augmented environment.

Probing from certain points of the three dimensional representations include the display of temporal behavior Figure 41 and the representation of higher dimensional parameters by exact numerical data or graphical representations on the panel (Figure 42). In contrast to probing, where information is extracted from the model, the pen can also be used to specify the origin of particles introduced into the flow (Figure

43). With the PIP all these functions are supported by only one device, which fits to the actual needs by changing its appearance.

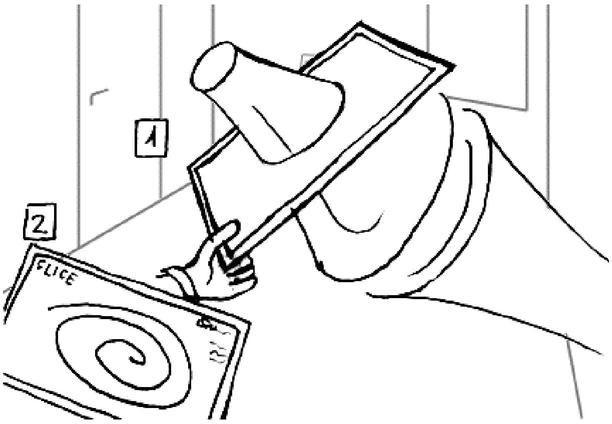


Figure 40. For scientific visualization the PIP can be used to specify and edit cutting planes …

Figure 41. … or measure simulated parameters at given locations and show instantly their evolution as 3D-graph on the panel.

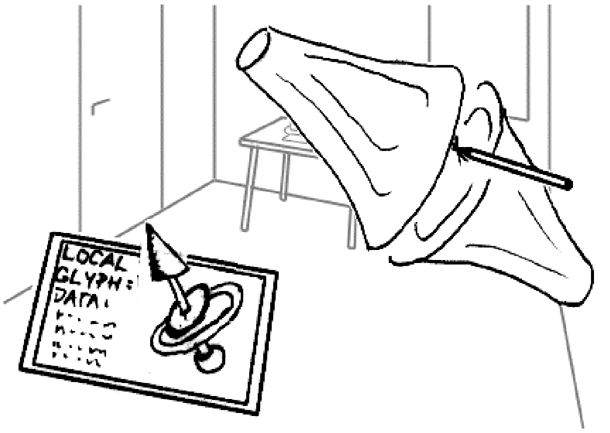


Figure 42. Multidimensional parameters at any point are shown using glyphs on the PIP or directly at the measuring point.

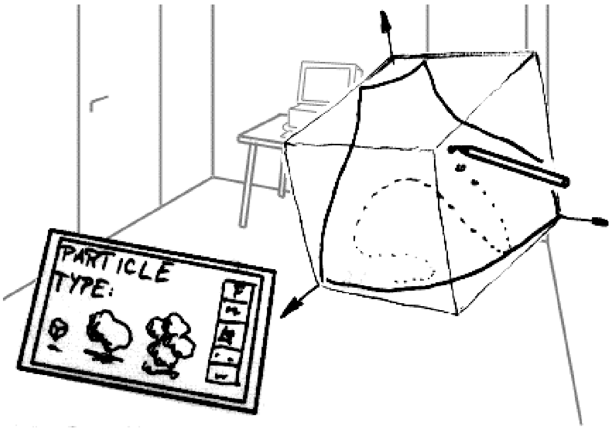


Figure 43. Introducing particles directly in an ongoing dynamic simulation should be very intuitive.

3.5.8 Organization of Interaction Elements

The exhausting list of different functions that can be implemented with the PIP interface is an argument for itself to create an organizational structure for these functions. Some metaphors showed that the parallel access to different tools or objects in a desktop-like manner could be useful. Considering the basic design guidelines we went back to the basic metaphor of the Personal Interaction Panel. The panel is a representation of a notebook that we use to write on. When a sheet is full we advance to the next page. In files containing a large number of sheets tabs are used to have quick access to a specific sheet that is located somewhere among other pages.

We abstracted from this real world example the concept of *sheets* on the PIP. Sheets are used for the organization of interaction elements on the Personal Interaction Panel's surface. Every sheet can hold one or more interaction elements that belong to an interaction task or scenario. Sheets can be accessed with *tabs* that are located at the border of the PIP (Figure 44). By choosing a tab a different set of interaction elements or controls appears on the panels surface. This change from one sheet to another can also implicate an application-switch inside the augmentation.

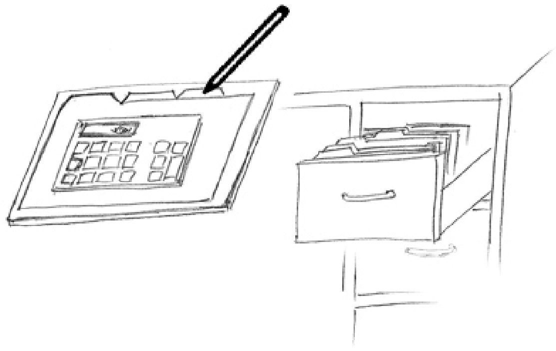
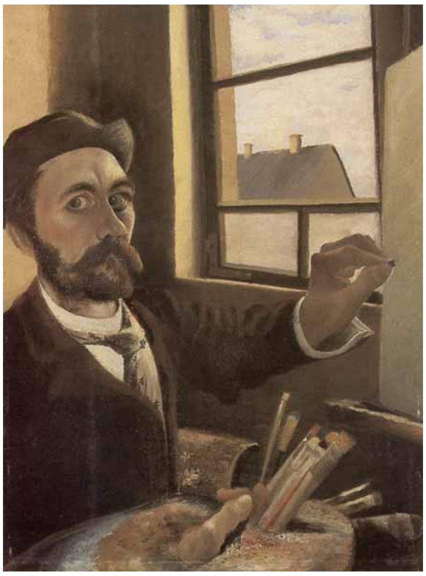


Figure 44. The concept of sheets and tabs helps to bring order into the chaos.

The opposite way of sheet activation is also possible, which means that an application or a selected object can *project* his interface on the panel's surface. In mobile versions a sheet-switch induction caused by a physical location change would allow to implement spatially aware interfaces like in [Fritzmaurice, 1993]. This high flexibility allows placing all interface elements onto the PIP, so that the interaction tasks can benefit most from the interaction device features. The concentrated *place of interaction* finally also helps to avoid scene clutter, common to many AR and VR systems.

3.5.9 Summary

The presented concepts for interaction metaphors with the Personal Interaction Panel show a wide spectrum of application scenarios, where and how the interface can be used to interact with virtual content. The physical asymmetric combination of a panel and a pen causes the device to support both two- as well as one-handed interaction metaphors. Altogether 12 degrees-of-freedom not only allow implementing complex natural interaction, but also require this comprehensive concept phase to refine initial ideas and avoid design flaws. The next chapter describes how hardware and software components of the Personal Interaction Panel were constructed based on these theoretical results.



Csontvári Kosztka Tivadar – Selbstbildnis.
(1896)

Chapter Four - Implementation

4.1 *Hardware Set-up*

During the research work a number of prototypes were implemented both regarding hardware and software (Figure 45). The following short overview should mainly portray the engineering process that was involved in the construction of this new interface.

In most existing implementations the panel and pen is tracked with *Polhemus Fastrak* six degree-of-freedom tracking sensors. Also the HMD we use to overlay graphics onto the real environment is equipped with an electromagnetic receiver (Figure 46). The base unit of the tracker serves the electromagnetic emitter and processes measurements of four receivers at 30 Hz each (we used the fourth receiver for another HMD). The unit is connected to a *tracker server* via serial connections, which is implemented on a dedicated Linux PC.

Figure 45. The physical setup of the Personal Interaction Panel.

Figure 46. HMD and props equipped with a Polhemus Fastrak electromagnetic receiver.

The *tracker server* sends position and orientation data using TCP/IP protocols over an Ethernet network connection to a multicast group. To receive the measurement data rendering clients may connect to this multicast group independently. This architecture is shown for the gaming scenarios in Figure 73.

During current research work rendering clients changed all the time, as rendering performance of workstations got better and better. However, for the first implementations three-dimensional rendering was done on a *Silicon Graphics Indigo2 Maximum Impact* computer. Later we used *Silicon Graphics O2* workstations, currently *SGI Octane* and lately also PCs are employed for rendering. All software implementations were developed in C/C++ using the *SGI Open Inventor* graphics libraries [Strauss, 1992]. Open Inventor is an object oriented 3D high-level graphics library based on a scene-graph approach.

Figure 47. *Virtual I/O i-glasses!* head-mounted display with a mounted Polhemus Fastrak receiver.

The resulting image is superimposed onto the real environment using a *Virtual I/O i-Glasses!* head-mounted display (Figure 47). This brand of HMD was very low priced, as it was targeted towards the mass market. It has a resolution of 263x260x3 for each eye and supports line-interleaved stereo video modes. The *field-of-view* (*FOV*) is 30 degrees diagonal, which is quite narrow for a good immersion. Unfortunately the company producing these glasses went bankrupt and only remaining stocks of the glasses are available.

One year ago *SONY* introduced the new *Glasstron* series of see-through head-mounted displays in a monoscopic version, this year the stereoscopic model was presented [Glasstron, 1999] (Figure 48). The appearance of this device is a new opportunity to get HMDs with a good price/performance ratio. This is one crucial factor for the spreading of the interface into real world applications. The Glasstron series has several different models, which differ in resolution and the capability to display stereoscopic images. The model LDI-D100BE is a high-resolution model (800 by 600 true RGB-pixels on each eye) supporting field interleaved stereo RGB-signals. Unfortunately this device has also a narrow field-of-view with 28 degrees horizontally as it was designed for watching TV and DVD-films on the road.

Figure 48. *SONY Glasstron LDI-D100BE* stereoscopic head-mounted display.

The Personal Interaction Panel props had during the time of this research work different manifestations that will be described generation for generation.

4.1.1 Generation 0

Right after the first theoretical inspiration we wanted to examine our idea very quickly. For the panel we used the *rear of an A4-dossier* that was cut off, and equipped it with the 6 DOF receiver using velcro fastener. A free *plastic pointer* from a box of transparencies (acknowledgments to 3M) was equipped with another receiver using the same "sophisticated" method (Figure 49).

The euphoria of the first successful Personal Interaction Panel in hardware was great, as the concept could be at least be examined in reality. The underlying software was an application to display a scientific visualization of the Wonderland model [Gröller, 1996] above the surface of the PIP, imitating holding it in the hand. Also two buttons were placed on the panel to switch between different variants of the visualization.

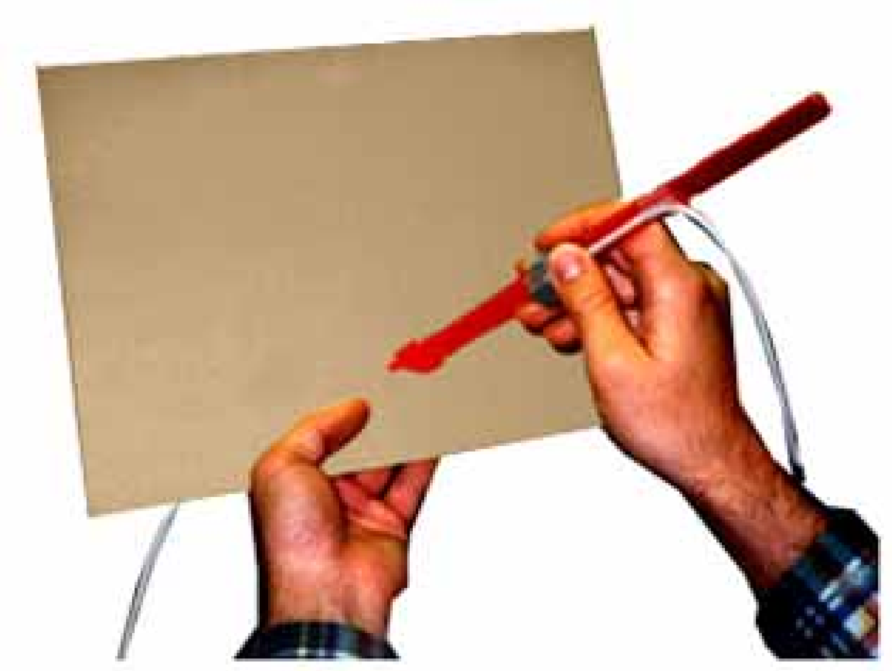


Figure 49. *Generation 0* of the Personal Interaction Panel interface.

The happiness of the first results faded while looking critical at the device. The rear of the dossier was not rigid enough and the fixing of the receiver turned out to be too loose, both causing registration-problems. The yellow color of the dossier was also chosen unfortunate. The overlaid graphics was tinted and had a small contrast to the yellow background. The *pen* was also completely passive, only the offset from the mounting point to the tip of the pen resulted in a position we used as *hot spot*. Grabbing of objects or any other mode change was not possible without at least one button on the pen.

4.1.2 Generation 1

Learning from the above faults the next hardware together with Fuhrmann we built had a more careful design. For the panel we used a lightweight 4mm thin chipboard (30 by 20 cm), which we painted black for better visual properties. Generation 1 of the PIP interface is presented at the beginning of this chapter in Figure 45. Horizontally centered, vertically 4 cm above the middle line, on the back of the panel we glued a small piece of plastic on the board (Figure 46). The receiver was mounted on this shoe using small plastic screws. This special non-metallic design is necessary not to interfere with the electromagnetic field used for tracking. The pen is made of 20 cm long 15mm diameter plastic tube, the receiver is mounted at the back end of the tube. It is equipped with a button at the tip and two on the side, convenient to operate them. This setup turned out to be very successful, serving for a long time as test-bed for a number of experiments.

4.1.3 Generation 2

After a number of encouraging applications, the Personal Interaction Panel was incorporated in the Studierstube-system. With the installation of the new Studierstube at the Institute of Computer Graphics in Vienna two new PIP interfaces were constructed. This time the panels were made out of black Plexiglas for a more robust design and professional look. Pens are similar to the preceding generation; mounting of the receiver is changed to be more stable. In this system an Ascension Flock of Birds trackers are used to obtain position and orientation of props.

Working with the new panel users reported fatigue after some period of time when holding the panel. The Plexiglas panel turned out to be too heavy for a longer time of use. However looking more stylish and being more stable this change did not advance the design. In future other materials and hardware elements will be investigated when building new items, up to now the chipboard seems to be the best solution for building the panel.

4.2 Design of the Virtual Part

Completing the hardware setup, the virtual part of the PIP interface was designed to fulfil ergonomic considerations and be as flexible as possible to conform different application scenarios.

4.2.1 GUI-Design

The first prototype application had a very simple graphical user interface (GUI) as described above. Only two buttons were placed on the panel's surface to switch between different versions of the model. These buttons were modeled as flat blocks, triggering an action whenever the hot spot of the pen entered them. Using the yellow dossier as background the color and contrast of the displayed elements was far from optimal (Figure 39).

Both theoretical as well as technical issues influenced the further implementation prototypes. Following list gives a summary of our ideas and experiences while implementing applications for the PIP interface. Considering these issues can help future PIP applications to benefit from a number of experiments.

◊ The size of the real panel was chosen to be familiar (about the size of an A4-sheet) and allow enough room for interaction elements. The virtual panel has exactly the same size as the real counterpart and is registered with it in a range of 2-3 mm correctly. Registration is completely static but is surprisingly sufficient good enough to allow precise operations with the elements on the panel as long as the user is in the guaranteed range of the tracker emitter. However the quality of measurements decreases rapidly when leaving the surroundings of the emitter. Noise ruins the measured data in some cases so much that subsequent position measurements are distributed around the real position in a sphere with a diameter of about 2-3 cm causing *jitter*! This fact influences size and design of interaction elements as described later. One of the major goals of the cooperation with the ICG at Graz University of Technology is to increase this accuracy using hybrid tracking technologies as described in [Auer, 1998] and [Auer, 1999].

◊ Users hold the panel with the non-dominant hand and interact with the pen held in the dominant hand. Using see-through HMDs the virtual part of the panel is always in front of the real counterparts. While the non-dominant hand is in front of the panel to select an element the virtual representation of the panel hides his hand and the real pen. The fact that users cannot see their hand while interacting was not noted as being disturbing, because this way the interface on the panel is always visible, however can conflict depth perception issues. This observation induced that it is also important to have a virtual representation of the pen in this see-through situation.

◊ Experimenting with the first implementation of the PIP hardware we found that a virtual representation of the panel itself is important, as contrast of the interaction elements alone on the yellow dossier was poor. We designed this background to have a convenient color and be in a better contrast with the elements on it. However not to distract attention from other parts of the augmentation and to be better integrated into the real environments we choose simple and low saturation colors. With following generations of the PIP, where the panel was painted black, this problem diminished, however the flexibility to control the background color using the virtual presentation turned out to be useful. Especially in gaming applications, this ability gives much freedom in the optical design of the game, as described in chapter 5.5.

- ◊ Orientation of the panel plays no role in the mean of tracking. Although the underlying metaphor of the notebook is vertically oriented we recommend using the panel horizontally oriented. Due to evolution, the human visual system is more landscape oriented and feels more comfortable with a higher aspect ratio. This causes also the trend to wide-screen television formats with an aspect ratio of 16:9. But there is another good reason using the PIP in the recommended manner: The employed *i-glasses!* head-mounted displays have a narrow field of view at an aspect ratio of 4:3. To see the whole surface of the PIP interface users have to hold the panel at approximately 40-50 cm in front. Using an upright panel would increase this distance to an inconvenient value. Unfortunately the *Glasstron* visors have about the same narrow field-of-view and are also at the same aspect ratio. HMDs with better FOV values – in this case especially in the vertical direction – would allow using a vertically oriented panel.
- ◊ We found that it is useful to have an approximately 2 cm wide border along the edge of the panel, modeled in a different color, to differentiate between an active region where interaction elements will be placed and an inactive region where users can hold the panel. Later we used this border at the top of the panel to incorporate *tabs* used to switch between different sets of interaction elements.
- ◊ During test implementations we considered also using the back of the panel, as this would double the surface to incorporate interaction elements. This feature was found cumbersome to be used in general, but as shown in chapter 6.3, it can be employed in certain cases.

The list mainly contains our reflections on the general GUI design of the whole PIP interface. These guidelines are not only based on theoretic assumptions illustrated in chapter 2.4.5, but involve also the results of a long engineering process. The following section describes the evolution and the palette of available interaction elements in more detail.

4.2.2 Interaction Elements

Following the basic design guidelines for AR interfaces the interaction elements on the Personal Interaction Panel were directly abstracted from real world interfaces. However most of them are today also part of conventional 2D desktop GUIs, we found it important to look at the real sources of abstraction.

When looking at our surroundings we will find a lot of knobs, dials, handles and different other elements we use to control devices in out everyday life. We are used to operate them and are familiar with their functionality. The interaction elements should benefit of these skills in the best possible way. For this reason the design of the elements was oriented rather on the real controls than on the 2D counterparts.

Buttons

First a series of buttons was implemented, which had different behaviors, found in real counterparts (Figure 50). An important basic rule for all buttons was to notify user on successful operation as the precision of the hardware could cause lapses.

◊ *Highlighting button*. This type of button highlights itself whenever it is touched with the pen.

◊ *Text button*. To identify functionality an inscription can be used placed on the button. By pushing it this can turn into a different text acknowledging the action, e.g. "OK" or a check mark. Because the insufficient resolution of HMDs we try to avoid the display of text in 3D whenever possible, and therefore this button type as well.

◊ *Shaped buttons*. Shape of the buttons can be used to indicate their functionality without any additional information. We implemented a series of arrow shaped buttons, which we used to cycle through a list of objects or scroll a map into four directions.

◊ *Animated buttons*. To resemble the original counterparts we implemented animated buttons, which animate themselves when touched. In the prototype implementation this behavior was fully scripted using *engines* in Open Inventor [Strauss, 1992] model files describing the geometry of the buttons.

◊ *Vicinity button*. This interaction element is not a button as found in reality, but from the implementation point of view it is a button. Using the same button code we can place invisible "buttons" in the 3D scene that react on the vicinity of the *hot spot*. We used this technology successfully for displaying a help message above the surface of the panel, indicating that the currently grabbed object will be pasted onto the PIP-clipboard when released. Although not being a real button this general method for noticing the proximity of the pen can be useful for a series of applications.

Compared to the bad performance of these elements in other VR and AR systems it is important to note that unlike in many other systems, interaction elements on the PIP are placed on a 2D surface in the 3D environment, allowing tactile feedback when operating the interface!

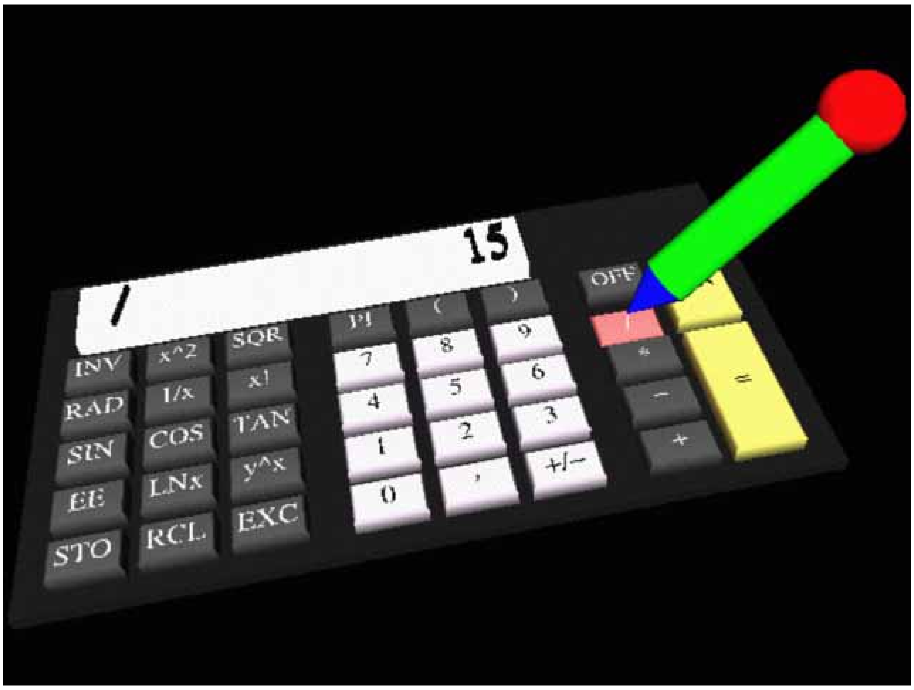


Figure 50. Buttons can be used to input numbers in this calculator application, written by Herman Wurnig.

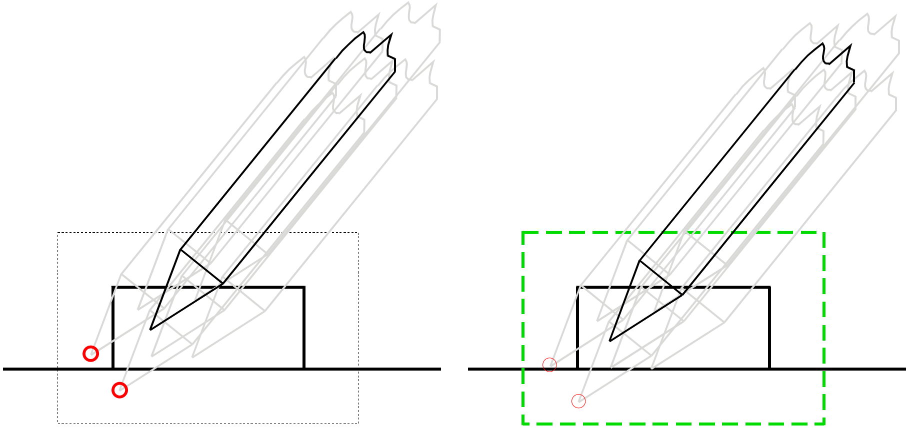


Figure 51. The introduction of the *spatial hysteresis* method helps to overcome problems of faulty triggering caused by noisy tracker data. Measurements marked with red in left image would cause retriggering of button. In the right image these points lie inside the spatial hysteresis.

As mentioned above we encountered the jitter caused by noisy tracker data to hinder the interaction with GUI elements on the PIP. The jitter in position data caused the *hot spot* of the pen to enter and to leave the button, while in reality the user retained the same position with the pen. This successive in and out of the hot spot triggered the button repeatedly, as in our implementation no other button-click with the pen's real button is necessary to operate the element.

To overcome this we introduced the method of *spatial hysteresis* for interaction elements. The idea is to cover the 3D-interaction element with an offset surface that acts as a buffer for triggering. When the hot spot enters the button geometry, the associated action is triggered. As a consequence of the built in hysteresis retriggering is not possible until the hot spot leaves the invisible offset surface around the element. For faster boundary checking in practice an invisible cube geometry is used for defining the space of hysteresis around the button (Figure 51). Using this method all disturbing effects of noisy position measurements of the tracker vanished. In sequence we applied the same technology to all of our interaction elements.

Sliders

To input other parameters than Boolean values we implemented sliders that allowed setting continuous values of application parameters. Sliders consist of a body that shows the interval of the parameter to be modified (Figure 52). In the first implementation the slide was activated whenever the hot spot intruded it's geometry.

The update of the slide position was immediate. Leaving the hysteresis geometry left the slide at the last position measured. User reported during tests, that they would rather "grab" the slide, modify it, and release for setting the desired value. Learning from this Wurnig modified the code in a way that the slide has to be "grabbed" by pressing the real pen button. As long as the button is pressed – even when leaving the hysteresis geometry – the perpendicular onto the slider is evaluated resulting in a new slider-value. This idea of modifying a value by "grabbing" the interaction element and performing the operation at a distance led to the idea of dials.

Dials

Dials are flat cylinders resembling potentiometer like gadgets, which can be "grabbed" by inserting the pen and pressing the pen button (Figure 53). Similar to the slider approach, manipulation is maintained until the button is released. The value is determined by projecting the hot spot position onto the 2D plane aligned with the base of the dial (i.e. the PIP's surface) and calculating the angle between the last setting of the dial and the vector pointing from the dial's center to the projected point. This allows to make rapid changes of a parameter when manipulating the dial near to it's axis and fine-grained modifications when the pen is further away.

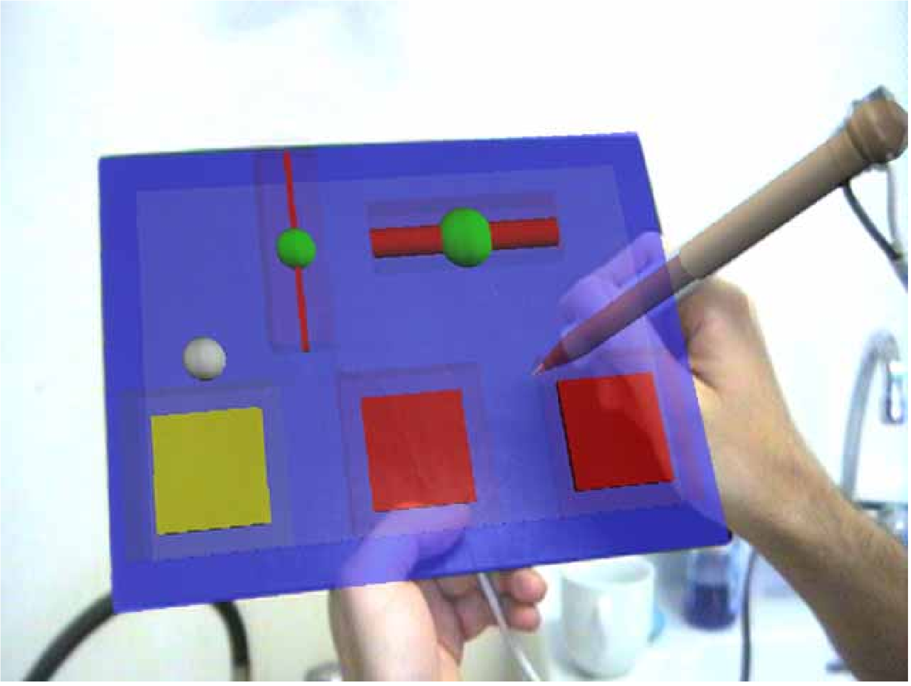


Figure 52. The panel with a selection of different buttons and prototype sliders. In this figure the *hysteresis space* around the buttons and sliders is made visible for illustration.

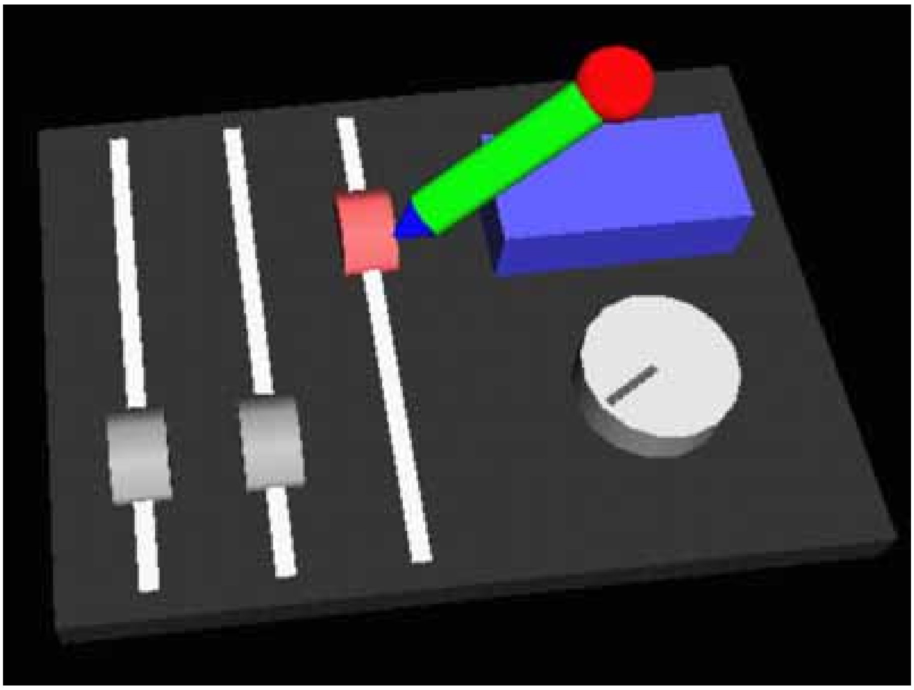


Figure 53. Dials can be used for rapid *and* fine-grained parameter modifications. This *color selector* allows setting RGB values with three sliders and the transparency with a dial.

Interaction classes

As a result of these element prototypes a new implementation for the Studierstube environment was realized by Fuhrmann and Wurnig. Following the object oriented fashion of Open Inventor a class-library was implemented, based upon Inventor's node kits. This allows easy integration into AR applications and makes software design more reusable. As a result of this work following entities were programmed:

- ◊ *Buttons*, radio-buttons and checkboxes (i.e. toggle-buttons),
- ◊ *Dials* with floating point- or integer-scale,
- ◊ *Sliders* with floating point- or integer-scale, and
- ◊ Sliders with logarithmic- or exponential (floating point-) scale.

These elements are now part of the *StbAPI* – the Studierstube application programmer's interface that is developed at the Institute of Computer Graphics in Vienna [StbAPI, 1998].

Using the described hardware setup, the user interface considerations, and the exhibited interaction elements a large number of applications was designed and implemented in very different application scenarios ranging from *Scientific Visualization to Augmented Gaming*. The next chapter gives an overview of these applications showing what we learned from different scenarios and how we transferred and applied this knowledge to other problem environments.

Professional writer – writes letters on commission.
(~1900)

Chapter Five - Application

Up to this point concurrent augmented reality systems and interaction devices were investigated, followed by a discussion of theoretical concepts about designing a universal tool for a wide range of interaction tasks. The conceptual description outlined a design draft for a hardware and software implementation of the idea. Now these drafts and prototypes are blended together to build application level solutions for the basic problem of interaction in augmented reality environments.

The first part of this chapter describes application fragments and applications in the same ordering, like the underlying metaphors were presented in the conception part of this work. This ordering is based on the three basic interaction classes of object manipulation, viewpoint manipulation and system control.

5.1 Object Manipulation - Scene Design

The fundamental task of object manipulation in a synthetic environment is crucial to almost all other interaction problems. Therefore we investigated a lot of simple applications in this domain to find out design-pitfalls.

5.1.1 Drag and Drop

First the pen alone as 6 DOF interaction device was considered to manipulate objects. The test applications we implemented supported grabbing, manipulating, and releasing – a simple *drag'n'drop* sequence – of geometric objects in the augmented scene (Figure 54). It is important to note here that already this simple test application confirmed the expected benefits of two-handed interaction. Users reported grabbing an object from the panel (in the bimanual frame of reference) to be an easy task even if objects were small, while grabbing objects from the scene (in an absolute frame of reference) was more demanding, despite of stereoscopic depth perception. With non-stereoscopic display this absolute target acquisition task was even more difficult to accomplish.

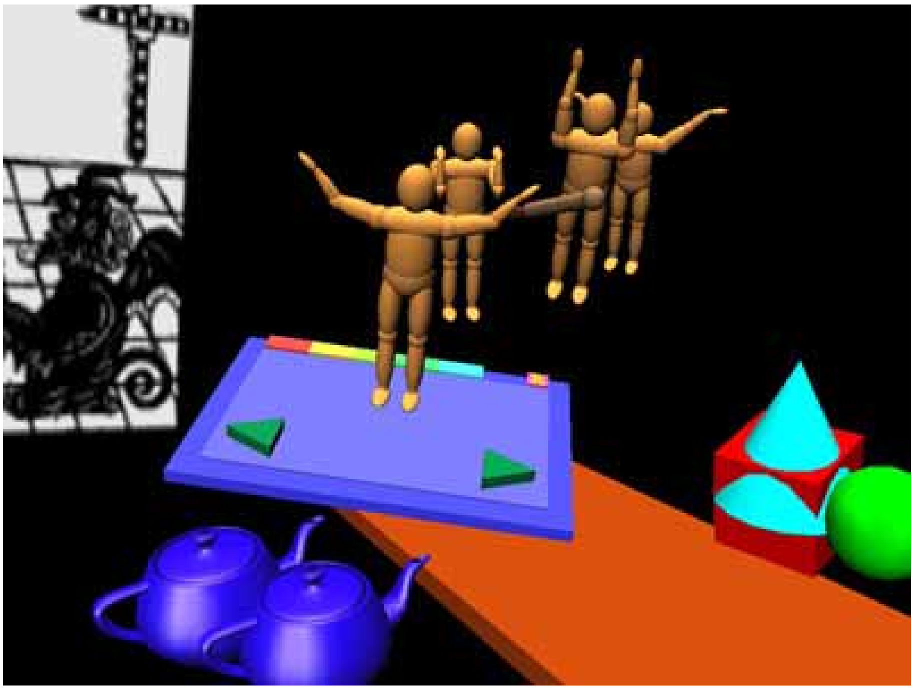


Figure 54. Scene design using *drag'n'drop*. In this and many other following examples the figures show only the augmentation overlay that is presented in a see-through HMD to the user. Black regions of the overlay let the light from real surrounding to pass through.

Positioning of objects over arbitrary distances can be achieved in one step, as the user can freely walk around the room in the tracker range. Rotating objects however sometimes required *ratcheting* – the subsequent grabbing, rotating, and releasing of an object – to reach the desired orientation. Again, as users can move free in the room, walking or turning around to the new orientation of the object can avoid this ratcheting.

A simple bounding-box check decided what to do whenever one of the pen-buttons was pushed. To make the built-in bbox-check of Open Inventor more efficient, we simply divided objects in the scene into different groups. *Inert objects* form the static part of the augmentation. In our case we modeled a wireframe model of our lab to visually check registration of the real- and virtual space. All these objects were stored in a separator node directly under the root node.

Non-stationary objects are collected under a special separator in the scene-graph that is checked for collisions when clicking the button. Provided that the check was successful the intersecting geometry is copied or moved – depending on which of the two buttons was pressed – to a separator below the tracker transformation node of the pen. After applying the necessary transformations from scene-to-world and from world-to-pen coordinate systems, the grabbed object seams to be attached to the hot spot – at the point where it was grabbed.

For the panel a similar separator was established under its transformation node to hold objects. A *vicinity button* cube covers the whole space above the panel's surface (about 20 cm high). When an object is released while this button was triggered (i.e. the user wanted to drop it on the panel and the pen is above the panel), the object will be moved from the pen container to the panel container instead of the scene container. To notify the user of the triggering of the *vicinity button*, the phrase "Paste here" appears on the panel whenever the hotspot is in the range of the surface.

This detailed description was necessary to understand the concept of *regions*, described in chapter 3.5.5. The described dedicated separators in the scene-graph are the regions, fundamental to *spatial controlled semantics* and the *privacy management* methods.

5.1.2 Precise and Constrained Manipulation

The simple *drag'n'drop* process is not always sufficient to manipulate objects in a virtual scene. The problem of most modeling applications in virtual environments is the alignment of objects placed into the scene. This results in messy scenes or in the worst case to faulty models. The precision of an alignment task is even more crucial in a gaming application described below, where game tiles have to be aligned to form groups. If the alignment requires too much concentration, players are distracted from the game itself.

In the mentioned gaming application game tiles had to be aligned to each other and to supporting surfaces like the table or the PIP holding the user's game tiles. These requirements effected the employment of *snapping* for object manipulation. Investigating the problem we found that we need only a very small subset of constraints. We use face-to-plane (binding three DOF) and face-to-face constraints (binding all 6 DOF) which are enough to place for example game-tiles in the board game. The face-to-plane constraint allows 2D translation and 1D rotation on the plane (see Figure 55 upper row), while the face-to-face constraint fixes the two faces onto each other (see Figure 55 lower row). In both cases the snapping condition for the automatic recognition process measures the distance and the relative angles between the faces. The distance is measured between the plane and the center of the face for face-to-plane constraints and between the center of the two faces for the face-to-face constraint, respectively. If both, distance and angle between planes are lower than a defined limit, the snapping condition is satisfied. To avoid jitter in the snapping, which results from noisy tracker coordinates, we applied a hysteresis on the snapping condition.

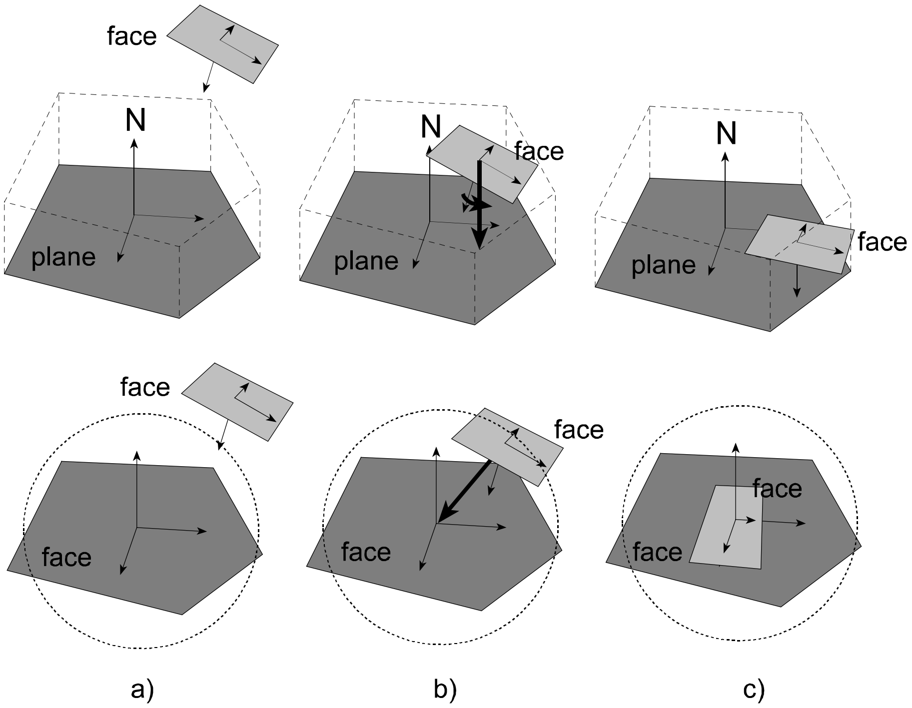


Figure 55. Face-to-plane snapping (upper row) and face-to-face snapping (lower row). A face is approaching a constraint region (a), moves into the region (b), snaps to the plane or face respectively (c).

In environments with many potential snapping-faces still a high computational effort is needed to find the nearest snapping face pair. However the constraint-recognition algorithm must be very efficient to be used as real-time application, so we used an axis-aligned bounding-box hierarchy for each, the static part of the scene, the geometry on each PIP and the dynamic part of the scene. This algorithm is similar to the broad phase algorithms like [Gottschalk, 1996] or [Cohen, 1995] for collision detection. For a small number of dragged objects (our test gaming application supports up to four players and each user has just one pen) we can achieve real-time performance.

The result of the constraint recognition is a list of valid snapping face-pairs. For simplicity we process only one snapping action at a time. We choose the nearest face pair (i.e. having the smallest difference in angle and distance) to perform the according transformation to the dragged object. This kind of priority selection behaves also very naturally because objects snap only to the nearest objects as expected by the user.

This technique increases both precision and task completion times. Informal user tests showed that concentration is not distracted by game-tile manipulation from the game itself. However the players were surprised in the beginning about the "stickiness" of the surfaces where game-tiles could be placed. Especially in the case of the table (a world fixed surface in the scene registered with a real tabletop) we had to fine-tune the snap condition (i.e. the distance where object snaps to the surface), as this was necessary for arbitrary placement and manipulation of game-tiles. However implemented in a gaming scenario first, the object-oriented implementation of this snapping mechanism allows other applications to benefit easily of this *precision enhancing technology*.

5.1.3 Object Browser

A simple yet effective demo application is the *object browser*. A list of objects defined in an Inventor scene-file is loaded on a special PIP-sheet. Using two arrow-shaped buttons the user can cycle through the list and explore the objects. Parts can be grabbed and placed into the surrounding environment. Figure 54 shows browsing through a list of objects. The same concept can be utilized to select tools from a list of possible candidates. Additionally a *Virtual Trashcan*, where unnecessary objects can be discarded, was implemented as a vicinity button (Figure 56). User notification is in this case the automatic opening of the cover.

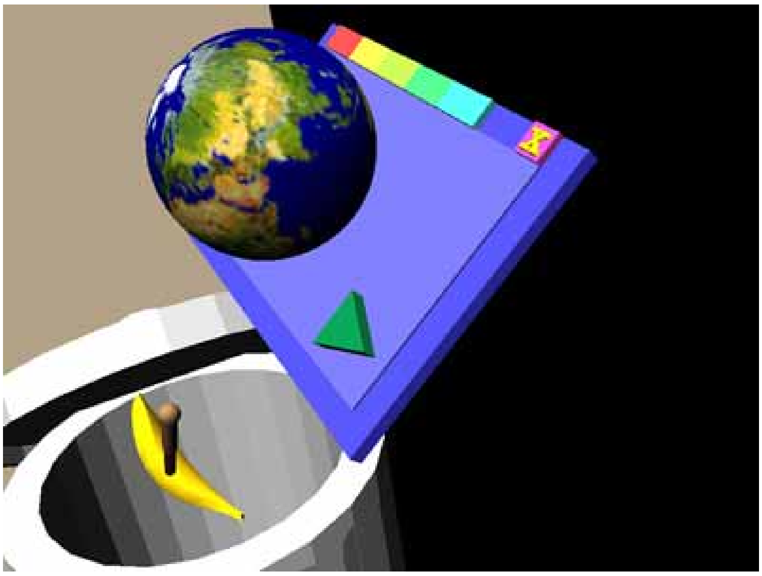


Figure 56. *Virtual Trashcan* – Users may delete objects from the scene by disposing them.

5.1.4 Room Designer

Unifying all above elementary tasks an existing Open Inventor based *Kitchen Designer* application of Erik Eckstein was adopted to work in an immersive augmented environment (Figure 57). An empty floor surface is displayed in the beginning in front of the user. A list of elements, that are needed to construct a kitchen are arranged on the panel. Using a *wall-drawing* tool walls can be directly *drawn* into the scene in 3D! With standard Inventor manipulators these walls can later be moved or resized. Items – like cupboards, chairs, etc. – can be *drag'n'dropped* from a list and placed into the room. Face-to-plane snapping aligns objects on the floor and constrains manipulation to the 2D surface of the ground. Multiple constraints (e.g. floor, wall) help to place objects as desired with a high precision. A special tool helps to measure distances in the floor plan.

Figure 57. *Room Designer* – a complex 3D modeler for rapid prototyping kitchen layouts.

This rather complex application showed that designing directly in 3D is possible with a high accuracy. The precision enhancing tools support rapid prototyping and so the design process is rather a process similar to equipping a dollhouse than a technical sketching task. In a multi-user setup like Studierstube this design process can be shared and discussed with other users.

5.2 *Viewpoint Manipulation*

Additional to the continuous update of the point-of-view of the user using head tracking for the head mounted display PIP applications also maintain further views onto the virtual part of the augmented environment. Using additional viewpoint manipulation techniques for navigation (i.e. modification of position and orientation relative to the virtual surroundings) is especially helpful in large VE applications. However in augmented scenarios navigation can destroy immersion, as the virtual part of the scene is normally registered with the real surroundings. Therefore the following map based techniques support rather a VE than an augmented setup. The two other metaphors implemented – *photo-camera* and *seams* – however proved to be helpful in AR systems.

5.2.1 Map

We implemented a simple *map-based* navigation metaphor for traveling though an infinitely large fractal environment (Figure 58). The dynamic terrain model provided by Schmalstieg was displayed right under the user's feet giving the impression of standing on the terrain. The terrain model was initially developed for a large-scale distributed environment [Schmalstieg, 1997]. However the scale for displaying the model was rather set to have the impression being a giant in the synthetic terrain model. In the application a minified version of this terrain is displayed on the panel's surface. The map is aligned to the (synthetic) north and is positioned so that the user is always in the center of the map (a true *egocentric world*). Four arrow-shaped buttons around the map can be used to scroll the map. Simultaneously the surrounding terrain scrolls under the user's feet correspondingly.

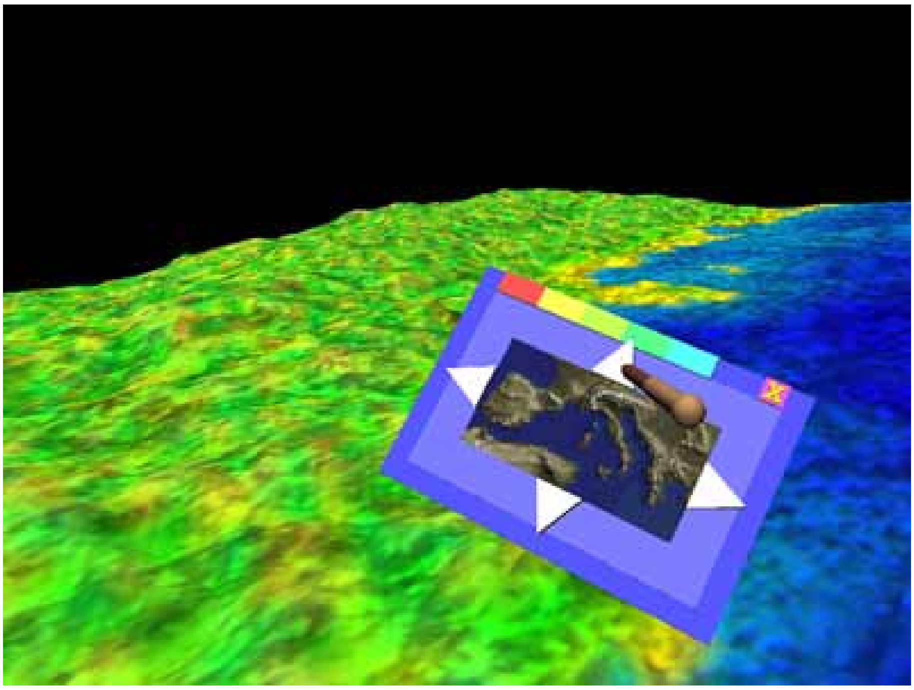


Figure 58. Map-based navigation gives two perspectives simultaneous onto the same virtual environment.

This technique provides two point-of-views onto the same virtual environment, supporting better way finding in a virtual environment. Users testing the applications reported sometimes to be confused by the fact that the map is always oriented in the same direction on the panel regardless to their orientation in the virtual environment. Shortly after our testing Edwards and Hand reported in [Edwards, 1997] using a maze running task the same problem. Their conclusion resulted that subjects preferred a view-aligned map that rotates according to the orientation of the interaction device in the environment. However other empirical evaluations [Darken, 1993] demonstrate that a view-aligned map is more effective for exploration, whereas a fixed map maintains a more consistent cognitive map of the overall environment.

In a future study we would like to perform the same comparison incorporating also our other concepts of map-grab and avatar-grab. This study concentrating on navigation in virtual environments is beyond the scope of this work, however in [Faisstnauer, 1998] we laid the foundations for this experimental study with the implementation of the *Mapper* – an intermediate module between input device drivers and virtual environment applications.

5.2.2 Photo-Camera

Navigating back into the augmented reality domain we set up a *Virtual Photo Camera* on the Personal Interaction Panel (Figure 59). This special PIP-sheet modifies the appearance of the panel's background it replaces the body of the panel with a transparent plane. This gives the impression, that the PIP itself is transparent, virtual objects can be seen through it. Also two buttons were placed onto the panel. When pushed the right button *fixed* the image on the panel providing a snapshot from the actual point of view. This snapshot is a texture that is rendered in an off-screen buffer and applied to the transparent plane on the PIP. We experimented with different methods for evaluating the point-of-view of the photo. Looking through the plane and fixing the current image turned out to be sometimes difficult in the mean of aligning the three components object – virtual camera – head onto the same line. Instead we decided to *shoot* the picture perpendicular to the panel's surface with a rather wide-angle lens. This method allows taking pictures in arbitrary directions similar to the famous *LOMO-movement*, e.g. the user can hold the panel high over an object and press the button to make a bird's-eye perspective picture.

Figure 59. The *Virtual Photo Camera* application lets users take pictures of virtual objects in the augmentation.

The other button in the application printed the photograph of the virtual object for documentation purposes on a printer located in our lab. As the see-through HMDs allow users to see the real surroundings, one can go and pick-up the printout without leaving the augmentation.

In [Schmalstieg, 1999a] a *snapshot tool* was used to provide similar functionality. This tool however supports an even more sophisticated functionality. It does not render an explicit image of the scene when making the snapshot; rather it saves the current scene database in the momentary version. This snapshot can later be activated giving a history function found in many design applications. The important aspect here is that the snapshot is a 3D representation that can be explored like any other 3D scene by choosing a new point-of-view. The current displayed image is rendered with a technology called *SEAM* [Schmalstieg, 1999b].

5.2.3 Seams

Schmalstieg explored the possibilities of his general approach to open windows onto a virtual environment and supposed to investigate the use of *SEAMs* for interaction tasks. An example is shown in Figure 60. Applications in the domain of scientific visualization using the *SEAM* mechanism are described in chapter 5.4.

The *SEAM* – Spatially Extended Anchor Mechanism – approach was originally designed to construct large-scale distributed virtual environments by providing *portals* that can be passed to cover large distances within the virtual environment or to connect to a new part of the virtual universe. The method modifies the rendering pipeline by inserting additional passes, where rendering of the secondary scene behind the seam-polygon is accomplished.

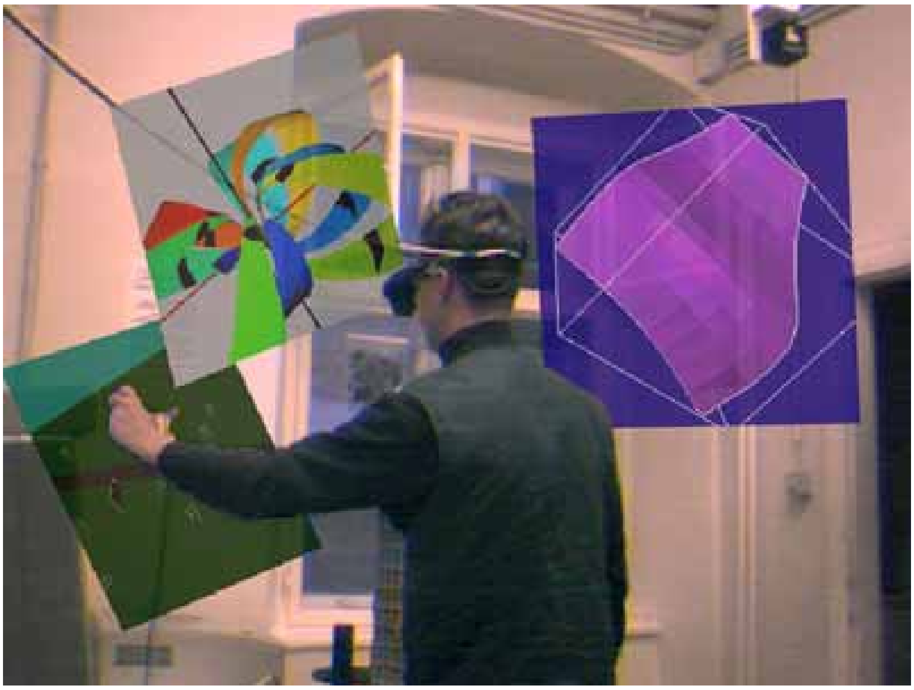


Figure 60. *SEAMs* for interaction. The user operates distant objects through a SEAM.

5.3 *System Control*

The set of user interaction elements to support *system- or application control* allows users to manage general tasks that influence the augmented reality system's operations. We review the implemented metaphors of *tabs* and the *application loader* shortly and progress then to higher level applications in the scientific visualization domain.

5.3.1 Tabs and Sheets

After implementing a series of demonstration prototypes we resembled our fundamental design goal to create a natural interface that is seamlessly integrated into the augmentation. For this reason *persistent augmentation is substantial not to break the immersion feeling* of the user. The staring of a new application in the augmented environment or system reconfigurations should be available within the environment to support this feeling.

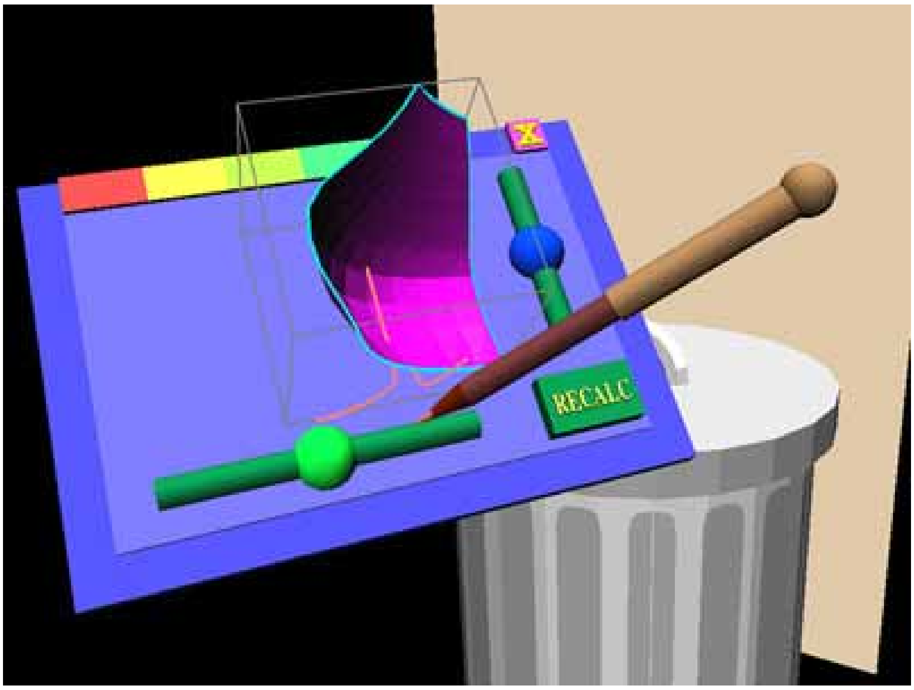


Figure 61. *Tabs and Sheets* – Tabs are used to switch between different sheets, holding a set of interaction elements for a specific augmentation.

To solve this we designed the concept of *PIP-sheets* containing application specific controls (Figure 61). In general one application is associated with one sheet. By introducing the *tabs concept* applications can be switched very easily. The augmentation is maintained to be constantly existing around the immersed user. However the last tab is used to exit the whole augmentation.

5.3.2 Application Loader

Tabs and sheets had to be pre-configured for a session in the augmented environment. The growing number of different applications and the need to maintain flexibility called for the implementation of the *Application Loader sheet* or also called *System sheet* (Figure 62). This special sheet is used to start any application from inside the augmentation. Invoking the sheet it scans the users home directory for applications written for the loader. A 3D graphical representation (3D icon) of the application is inserted in a list displayed simultaneously on the PIP. Clicking on the corresponding representation starts the application by loading and initializing the shared object library of the application. This makes dynamic runtime loading of applications into the augmented environment easy. This functionality is now integrated in Studierstube supporting it's Workspace concept [Fuhrmann, 1999].

Figure 62. *Application Browser* – 3D icons on the panel indicate found applications that can be directly started from within Studierstube.

5.4 *Scientific Visualization*

5.4.1 Studierstube

After detailed descriptions of Studierstube's features in chapter 2.2.2, here we focus on the implementation and the short description of scientific visualization applications in the Studierstube environment.

The original goal of Studierstube was to construct a visualization environment, where multiple users can work with 3D data, as if they were real objects. To achieve this when working with complex scientific visualizations Studierstube needs to be connected to a visualization system. Fuhrmann and Löffelmann presented in [Fuhrmann, 1997] how a collaborative scientific environment has been built using Studierstube and by employing the Advanced Visualization System AVS [AVS, 1992] as a *Simulation Engine*.

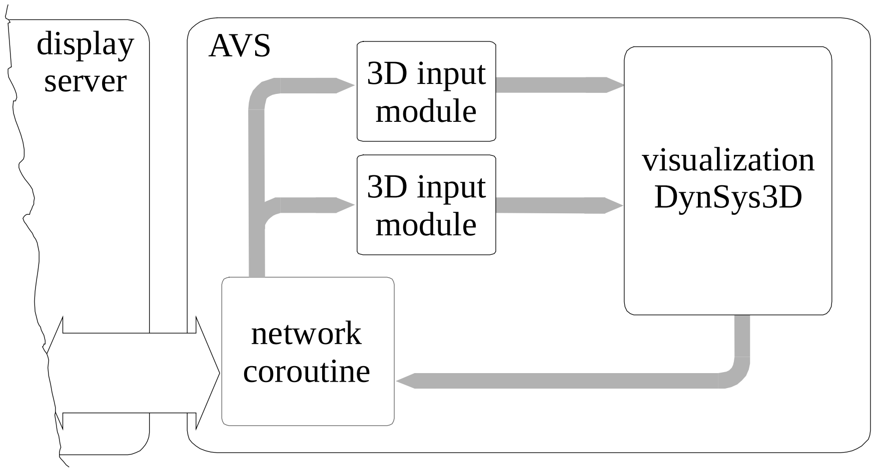


Figure 63. Integration of *AVS-DynSys3D* as *Simulation Engine* in Studierstube. Studierstube itself is represented by the term *Display Server*.

A loose coupling is defined between AVS as the computational back end and the visualization server that coordinates interaction with the model in the Studierstube (Figure 63). Visualization data is exported from AVS to the visualization server that takes care of distribution of the data among the Studierstube's clients. *Computational steering* – the process if interactive manipulation, observation, and discussion in scientific visualization – is achieved by using special input modules for AVS that accept new values for simulation parameters from the Studierstube.

Modifications of the visualized data that do not involve simulation calculations (such as rotating the simulated model) can be carried out in a close loop by the Studierstube system alone and do not pass data between Studierstube and AVS. Such simple interactions are not affected by the performance penalty created by invoking a complex software system such as AVS and can therefore always be carried out with real-time response and high fidelity.

5.4.2 Flow Visualization

Based on the first experiments with the integration the concept of connecting a *Simulation Server* to Studierstube was generalized. The multi-purpose workbench for the rapid development of advanced visualization techniques in the field of three-dimensional dynamical systems *DynSys3D* was connected with this link into Studierstube as described in [Löffelmann, 1997] and [Fuhrmann, 1997]. This system supports standard visualization techniques including streamlines, stream surfaces, and particles to illustrate the investigated systems (Figure 64).

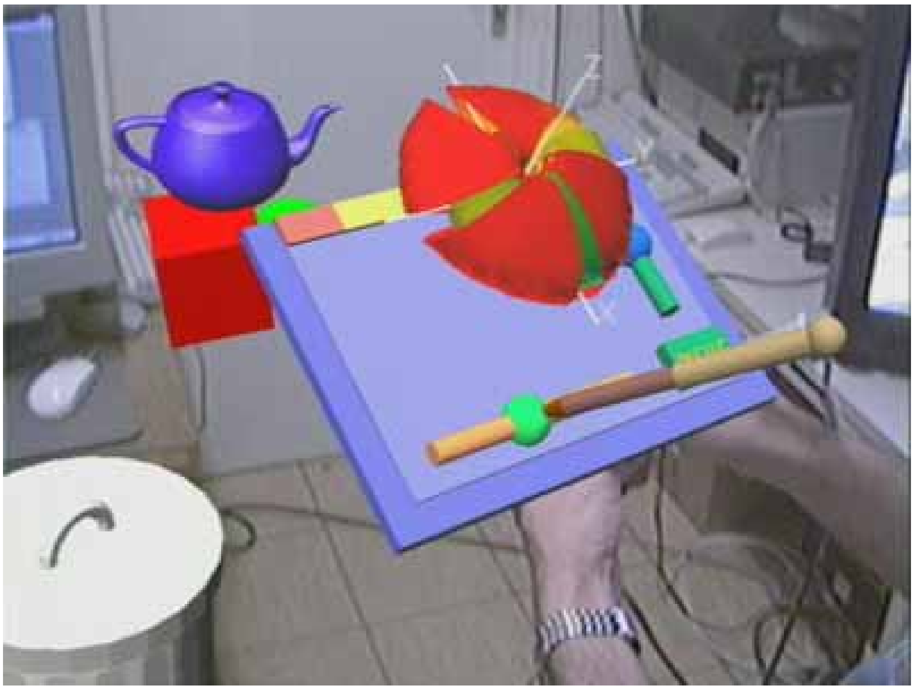


Figure 64. Investigating an *R-Torus* on the panel.

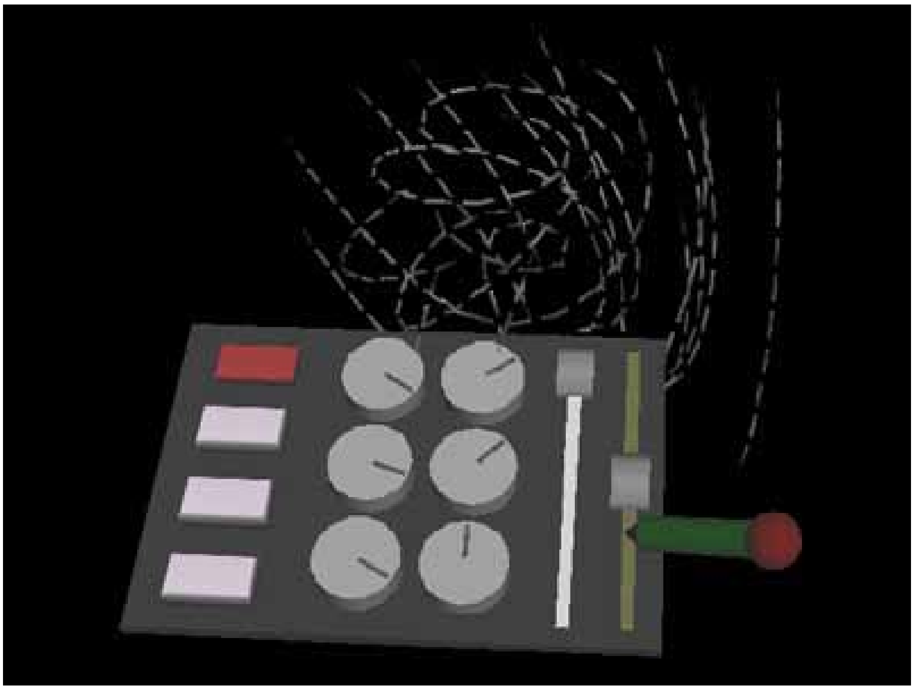


Figure 65. The *Viskas* system makes use of a large number of interaction elements to control parameters of the visualization.

The expressive power of the visualizations is enhanced by the opportunity for natural collaboration in an augmented reality setup. Furthermore, the PIP enables with a custom tailored interface for interaction with the dynamical system interactive probing and the introduction of streamlines and stream surfaces into the system. This type of direct interaction gives a natural feeling of handling visualization data.

Fuhrmann described in [Fuhrmann, 1998] a series of other visualization techniques – summarized under the code name *Viskas* (Figure 65). Here the PIP interface supports the application heavily with a number of interaction elements (dials, sliders and buttons) for parameter control.

5.4.3 Röntgen

We have found that SEAMs are a useful tool to display different layers of information, which is especially problematic in the case of 3D data. Using this tool different aspects of the same 3D database can coexist on the same spatial location visually not interfering. For orientation a basic structure of the object to be investigated can be displayed. The panel or the pen can be used for interactive manipulation of the SEAM-polygon. Held against the structural representation, additional layers of the underlying database can be displayed to allow a personal exploration. Wurnig used this approach to implement an *X-ray lens* similar to the *Magic Lenses* approach in [Bier, 1993]. The SEAM-polygon is attached to the tip of the pen to give the impression holding a lens (Figure 66). This lens is also a very good example of an *extended tool* that users found very natural to use with no explanation at all.

Fuhrmann extended this approach in [Fuhrmann, 1998] and built six-sided boxes of SEAMs to implement an interactive metaphor similar to the approach of 3D Magic Lenses in [Viega, 1996]. These boxes can be used to display higher resolutions or in other means different representations of the flow visualization investigated (Figure 67).

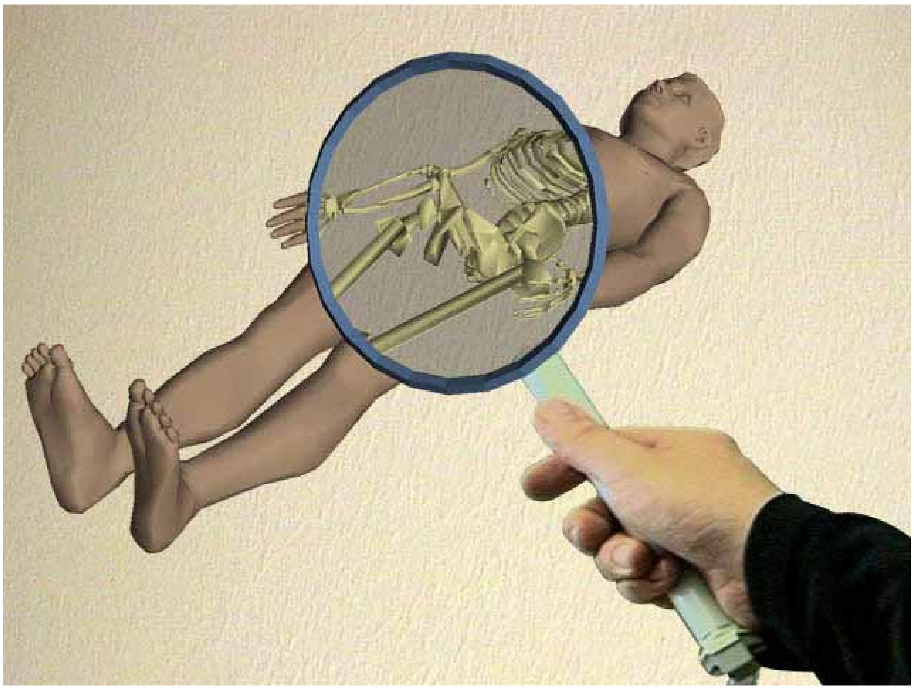


Figure 66. Röntgen Demo – Manipulating the lens users can interactively explore the 3D-bone structure of the model.

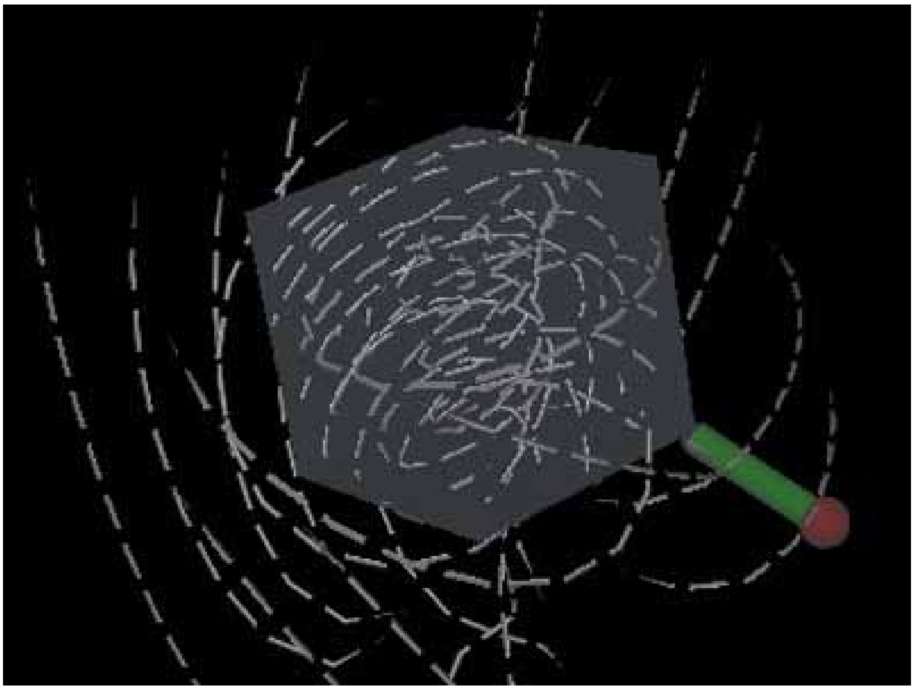


Figure 67. *Magic Box* – The user can browse different aspects of the 3D visualization using the box attached to the tip of the pen.

5.5 Collaborative Gaming

This topic seems to be radical in a thesis. Why should someone deal with gaming when concentrating on human computer interface design in a research work? Why is the game domain a good playground for augmented reality research? And why is augmented reality a good playground for gaming? Following discussion and the presented implementations will give answers to this list of questions!

5.5.1 Why Gaming?

Interactive gaming is becoming more and more one of the dominant application areas for computer graphics. The related industry is growing very fast both in the location-based entertainment (LBE) and the PC-game domain. We focus on a computer-based gaming environment for home usage, which is especially suitable for *multi-player board games*. We identified three major aspects that drive these types of games: *Social communication*, the freedom to maintain individuality in a *private space* and the *fast and precise manipulation of game-tiles*, which is important to maintain a dynamic course of game.

The *social communication* aspect can be clearly observed with non-computer based multi-player board games like Mah-Jongg, Trivial Pursuit, etc. Computer based games, which gather around some classical computer-game idea (like Jump-and-Run games), often fail to support this type of communication and brought criticism to the computer games.

The other important aspect is *privacy*. Parts of the "gaming space" like the table in a board game are public. Every user has the same visual access to this common information. Parallel to that the game has to maintain a private space for the player. This space provides security for individual strategic decisions.

Game-tiles form the backbone of a board game that is played in a group of users. These *tiles have to be manipulated quickly and precisely*, so that e.g. the action of placing a tile on the table carries only the message of a strategic step and not the effective physical manipulation of a piece of plastic. This keeps the course of game going on and does not distract attention from essential decisions.

These three aspects are weighted differently in different kind of games. In a game like Trivial Pursuit for example, the social communication has more weight than the private space. Privacy is only needed to hide answers from the other players, whereas Mah-Jongg needs a private space for each user to hide tiles from others during the game and needs a precise manipulation of the game tiles.

We identified the *ability to display different information* to each participant, *unhindered social communication*, and *precise and fast interaction* with the game as the crucial factors for augmented gaming. We describe a technology setup, which provides a good base for this requirements and present techniques to overcome limited precision in an augmented reality environment. The *principle goal* of this detour *is to obtain results that can be generalized* and be applied to other application domains. But before describing Mah-Jongg, our most complex gaming implementation we present some ancestors.

5.5.2 Palette of Augmented Games

Blockout

The first test implemented together with Greimel was a 3D-augmented version of the well-known *Blockout* game. This game with the goal to shoot out blocks from a wall with a paddle and a jumping ball was used to investigate the *proprioception* ability of users in a limited visual environment. The i-glasses! HMDs we use have a very narrow field of view, so it interested us whether it is possible to hit the ball without looking at is. The gaming scenario helped us to motivate users to concentrate on the goal instead of focus on the direct action of hitting the ball.

In our implementation blocks to be broken out were placed over a door in our lab, the ball jumped free in 3D and was reflected from all six walls and some registered furniture items like desks. The user held the pen in the dominant hand like the handle of a racket. The head of the racket was an augmented extension attached at the tip of the pen. The diameter of the racket is about 30 cm. The goal was to hit the ball with the racket towards the wall and to throw down all bricks to reveal a picture behind the wall. Ball behavior was simulated using a simple dynamic model. The augmentation overlay is shown in Figure 68.

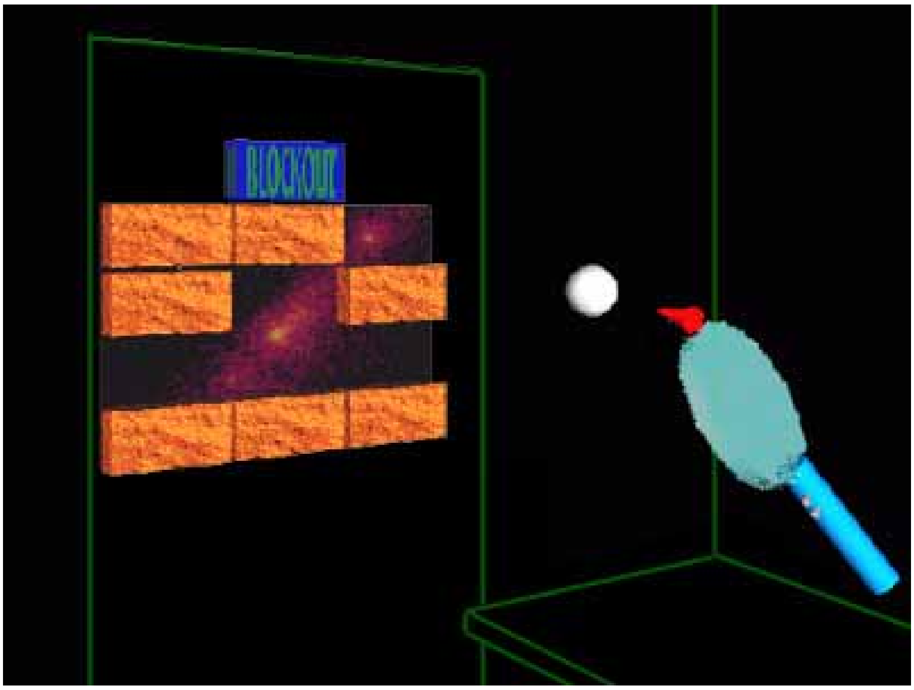


Figure 68. *Blockout* – The virtual ball is reflected from real objects in our lab represented in the overlay in wireframe. The goal was to throw off all blocks from the wall.

The game was proven to be quite dynamic, players concentrated after the first two- or three shots on the game itself. After a practice time of about 5 minutes users were able to play the game by just looking at the door with the overlaid bricks. As in real tennis gaming users did not look at the ball to hit it, they just used their proprioception to locate the ball and hit it back towards the wall.

Parallel to the prove of the good results on proprioception we also noticed following:

◊ When the user missed the ball, he or she had *problems of finding* it *again*, due to the narrow FOV of the head mounted displays. As the ball jumped around the environment – in some cases at a rather high speed – players had to scan the room again and again to find it. To lessen this effect we implemented a compass at the tip of the racket that pointed always to the ball. This aid helped a little, however when the ball was jumping at a high speed, it was still difficult to locate it. It would be interesting to investigate whether 3D sound could help to improve target acquisition in such an environment.

◊ During test we modified the dynamic parameters of our ball model. We observed that dynamic behavior close to real world dynamics is crucial for proprioception. When the ball was too slow or damped too much by the wall, users miscalculated the future position of the ball and missed it. The human senses seem to be adjusted and optimized for our real environment due to a long evolutionary process, virtual- or especially augmented environments should be carefully designed not to conflict with this, if subconscious user action is expected.

◊ A rather technical problem in the beginning was the *thickness of the racket*. As we checked the ball collision with a simple bounding box check against the racket, in cases of very fast movements the ball was missed. This proved the requirement of decoupled simulation in similar environments, where simulation updates must happen more often than rendering updates (of around 25Hz in our case) for a correct physical simulation. As the tracking system provided only 30Hz in the given setup there was no means of increasing simulation updates. Instead we enlarged the bounding box of the racket to be about 10 cm thick, so that the ball was hit reliable.

This very simple application resulted already in a list of very interesting observations on a one-handed manipulation task in a dynamic augmented environment. The following more complex applications can benefit from these findings.

Maze Game

In the second experiment we transformed the original Open Inventor demo maze into a two-handed game on the Personal Interaction Panel. We think that our implementation resembles the original game idea better, however that is manipulated with two dials at the side of a box as shown in Figure 69. Users can play the game seeing a stereoscopic version of the maze directly on the panel's surface. Our *Maze Game* turned out to be simple, yet very successful. Test persons – absolute VR or AR novices – reported the interface to be very natural as the right settings of dynamic parameters helped to suspend disbelief in this mixed reality situation. They were so much concerned about the game that they tried to cheat by letting the ball jump over the holes, (which was unfortunately not implemented).

Figure 69. *Maze Game* – The ball must be navigated through the maze by tilting the panel.

We have noticed that except some rare cases, users instantly held the panel with to hands, however it was light enough to be held by one hand without fatigue. When asked, they told to find it easier holding the panel with both hands, because it is more stable and they can finer control the manipulation. We accounted this observation to be a very good example to prove Guiard's results on bimanual action.

For comparisons we implemented also a version, where control was maintained with the tracked panel, but the output could be seen on a monitor with a fixed point-of-view in the virtual scene. Users liked this variant as well and found the indirection of the displayed image to a different position not to be distracting, despite our basic design guideline describing the need of correct registration. However this version of the Maze Game was not augmented in the traditional sense, it was so successful that we displayed a two-user variant in a setup with a large-screen passive-stereo projection at an exhibition in Vienna for the period of about a week (Figure 70). Hundreds of visitors played the game with no explanation at all. The interface was so simple and the interaction so obvious, almost free of any metaphor that introduction was not needed to play the game.

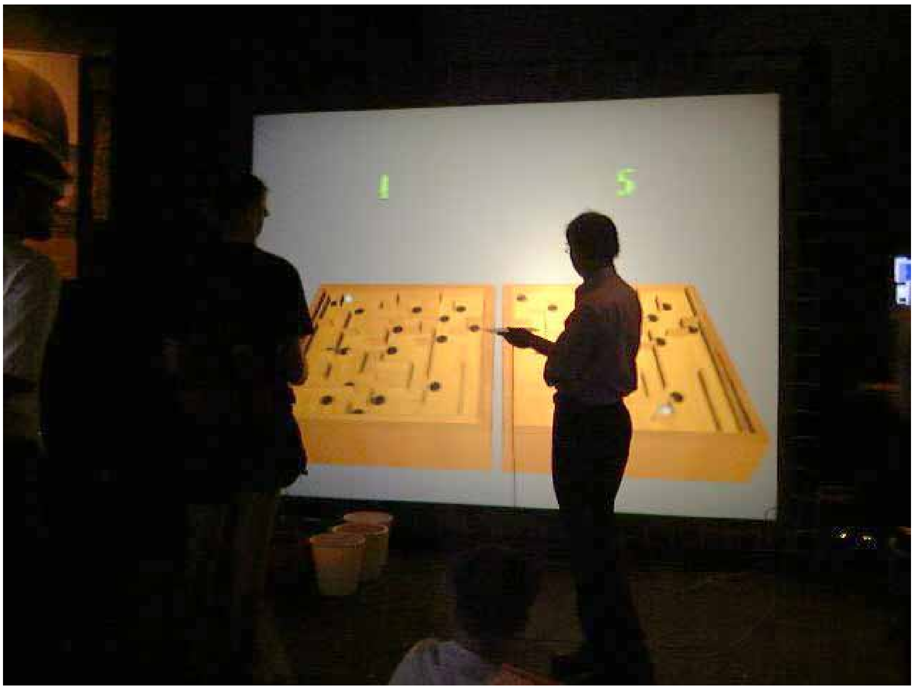


Figure 70. *Two-User Maze Game* at the Synworld '99 exhibition in May 1999.

Virtual Painting

With Sainitzer we used the same setup of passive stereo-projection for implementing a virtual *PaintStudio* similar to the idea in [Agrawala, 1995]. To make the usage of the input devices even more obvious, we exchanged the pens with regular painting brushes, which we equipped with a Polhemus electromagnetic receiver. In this game users had to paint the models of animals with virtual paint, taken from real paint buckets (Figure 71). Brushes changed the color by dipping into one of the (in reality empty) buckets located on the floor in front of the projection. We have chosen animals being familiar and easy to identify for different ages of visitors. Again no additional explanation was needed when users located the surface of the animal floating free in front of the projection. The passive stereo projection gave a good sense of depth for surrounding users.

Figure 71. *PaintStudio* using real brushes and virtual paint. Two-user can simultaneously paint the animal both cooperative or against each other.

Our observation at the exhibition with subjects of the age from 3 to (about) 60 – who were fascinated by the simplicity of the approach and have experienced a virtual/augmented environment for the first time in their live – resulted following findings:

- ◊ A natural interface or interaction device can make tasks very easy to understand. No additional explanation of the interaction is needed and users can transform their skills into the synthetic environment, whenever natural clues are not destroyed.

- ◊ The natural acceptance of the whole installation made simultaneous "painters" to cooperate or to be against each other (using an erasing paint the model could be rubbed out), even when players did not know each other.

- ◊ Users reported the fact that the virtual brush was not correctly aligned with the real brush. This caused sometimes confusion in the beginning. As the two users and the surrounding observers saw the same projection, it was not possible to maintain a correct perspective for everyone. Instead we assumed an observer standing about 2m centered in front of the projection. However, once players found the virtual animal in free space they could easily follow its surface.

- ◊ To ensure that the animal can be painted everywhere we inserted a rotor node into the scene graph for rotating the object. After several minutes users evolved so that they could also target the surface of the animal when rotating.

- ◊ Users maintained after a while a cognitive map of the stage. After dipping the brush into another bucket to select a different color, users went back automatically to a remembered position on the stage and started to paint.

Despite being again only a one-handed metaphor Virtual Painting produced a number of valuable observations we can transfer into the design of other applications.

Virtual Casino

A good example for the multi-user aspect of gaming is a casino roulette table with multiple players around it. The social channel is very important in this game. Because of the almost 50% mean-chance for a win, even with a small amount of players there is almost each turn a winner at the table, giving the whole game a positive mood. Playing the game alone would destroy this social aspect and make the whole game much less interesting. In this game individuality and private space as pointed out in the introduction is important.

Our setup of *Virtual Roulette* consists of a common roulette-table aligned with a real table in our lab. Players can place their coins onto this table Figure 72. The PIP is used as wallet to hold money individual for each user. Like in real world players can show or hide their coins by simply turning the panel away from the other player's view.

The game allows a couple of users to place their coins onto the table independent of each other. Coins placed on the table are bets that are subtracted from the players' wallet. Money placed on the table can be seen by every player, whereas only the owner can manipulate his own coins. Each user can turn on a *personal help* that is displayed to him individually without disturbing the others as users wear head mounted displays.

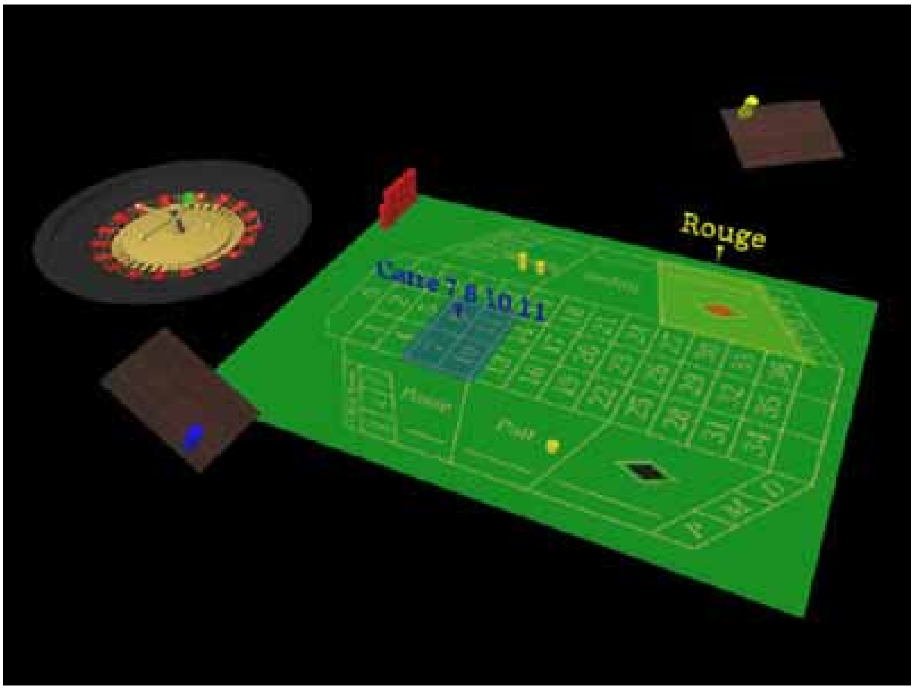


Figure 72. *Virtual Roulette.* Our implementation shows the table with multiple users playing.

The implementation of the game is based on a *Java-Server* and rendering clients developed by Sainitzer and Buchegger. The server distributes the scene-graph in a serialized form to rendering clients, where it is reconstructed for rendering. Position and orientation updates are obtained directly by each client through a multi-cast group using UDP messages.

Results of this collaborative setup for gaming were promising. The interaction metaphor was very simple. Coins can be placed by clicking the pen-button while holding it over the required field of the roulette table. Clicking the button in its vicinity starts the roulette-wheel. Although this game, mainly because of its gambling character, does not involve very demanding interaction tasks, it was our first collaborative gaming setup in augmented reality.

5.5.3 Game Testbed

To profit from our previous results we extended and generalized our gaming setup to support a number of gaming scenarios with the same hardware and presented it in [Szalavári, 1998b]. Our aim was to overcome the problems of previous approaches and construct a conceptual hardware environment for playing games as a highly demanding human-computer interaction task.

To achieve this we also describe how privacy management is implemented using a sophisticated design for automated security control.

The *setup* we have chosen is similar to that of *Studierstube* [Fuhrmann, 1997], consisting of private see-through head mounted displays (Virtual I/O i-glasses!) and a Personal Interaction Panel (PIP) [Szalavári, 1997] for each user. HMDs and interaction devices are tracked in position and orientation with a magnetic tracking system. The see-through HMD does not block the view onto the environment, so additional to verbal communication gestures of other players can be recognized. Using dedicated display devices for each user, display of private information is supported. The Personal Interaction Panel consisting of panel and pen allows powerful two-handed manipulation of virtual objects with high precision. Interaction with the game is independent from other users as the interface serves as personal tool.

The proposed *system architecture* consists of a *Game Server* maintaining the graphical database of the game and running the simulation based on the input data from magnetic trackers and input devices. *Clients* for each user render the customized view on the shared environment. Tracker data and information of the buttons on the pens are distributed over the local network using a *Tracker Server*. Using a multi-cast group every participant gets synchronized updates of tracker data for rendering. This approach allows scalability in the number of users to a certain extent. An overview of the system architecture can be seen in Figure 73.

The current hardware set-up as shown in Figure 74 supports up to four users playing simultaneously in an augmented environment. Rendering is done on a SGI Maximum Impact R10000 with Impact Channel Option (ICO). The ICO is utilized to split the frame buffer into four regions and to convert output to VGA signals. Using converters images are presented in the HMDs. Our implementation using Open Inventor can deliver independent line interleaved stereoscopic rendering for four players at about 10 frames per second.

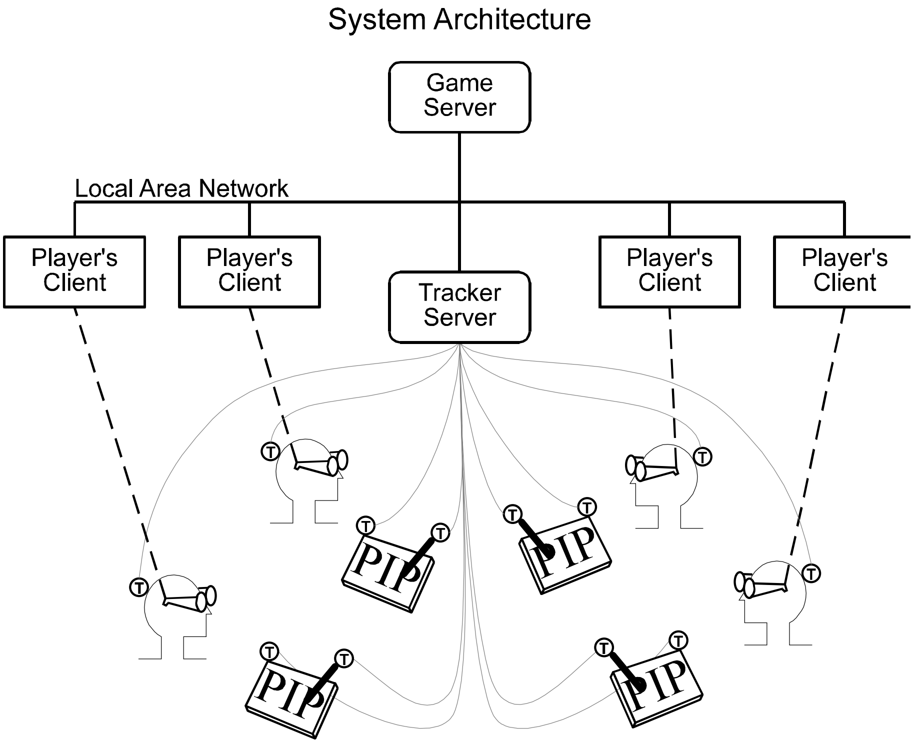


Figure 73. The client-server architecture of our system.

Figure 74. The hardware environment as seen from a non-participating observer. Each player wears a see-through head-mounted display and interacts with a Personal Interaction Panel.

5.5.4 Privacy Management

As pointed out at the beginning of this section *privacy* is a very important aspect of collaborative gaming scenarios. The use of private and public space provides security for individual strategic decisions as well as a common playground to communicate with other users. The method of *privacy management* we implemented can act as general solution to security problems in virtual and augmented reality environments; therefore a more detailed description of the implementation background is given here.

As described in chapter 3.5.5, security information controlling the behavior of scene traversal for different players uses presets we call *regions*, coding specific variations of the traversal for each user. This concept can be best described with a set of keys and locks as shown in Figure 75. The locks are associated with objects in the scene. Different keys are handed over during traversal of the scene graph to the different players, when entering the hierarchy associated to a region. Another dimension to our key system is added by the diversity of scene traversal actions, like rendering, picking, snap-condition evaluation, etc. Therefore we use complex keys encoding security for each action type, presented by the different edge-shapes in Figure 75 b) and c).

The method is implemented using Open Inventor and is integrated tightly in the Inventor scene-graph concept. We encode the necessary security information in *layer-nodes*, stored in the common scene graph, which also holds the whole graphical database. This concept allows easy replication to clients. To render this common description in a player customized version, only the player-id has to be set at the very top of the hierarchy. We encode security information in a 2D matrix for different players of the game and different actions. So unlike many other approaches privacy information is not stored as object property of each entity, but at a higher level as region property. Layer-nodes can be everywhere in the scene-graph. In this way is possible to form a hierarchical security structure. Sub-layers inherit rights from the super-layer. This allows defining group rights in an easy way.

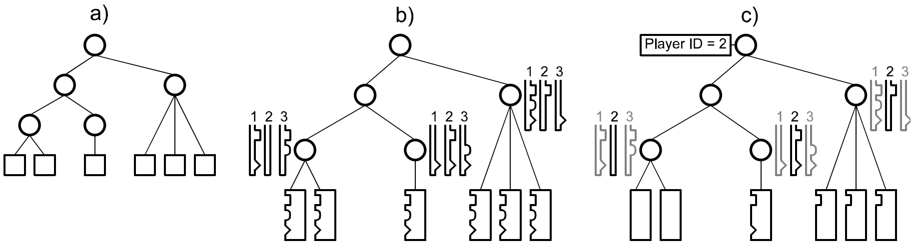


Figure 75. Representation of the scene graph. Figure a) shows the graph as the game engine sees it. Figure b) shows security information - presented as “keys” - in nodes belonging to different user IDs. Leafs hold sub-graphs, with general “locks” that represent different behavior during traversal. Figure c) shows how player 2 sees the common scene graph.

Finally, we extended the underlying system by components to interpret security information at any point of the hierarchy. Using information passed down in the hierarchy, sub-hierarchies, e.g. objects, or in a gaming scenario game-pieces, can react to the security information, behaving differently depending on the security setting at higher levels. Thus members of groups inherit security state, transition to another group (i.e. region) automatically causes an implicit state change as shown in Figure 75. However, our security layer concept is so general, that it can implement other security strategies. Storing security information on item basis by defining a layer to each object resembles previous work.

The system is open for dynamic modification and reconfiguration of security presets at runtime. Although we think that a careful application design with the identification of good security presets can avoid this and free up the application from additional security management.

The presented approach leads in our gaming scenario to a very effective and powerful mechanism for security management. On the example of Mah-Jongg we showed how this mechanism can help to transform a traditional game to augmented reality.

5.5.5 Mah-Jongg

A large number of different games seem to fit excellent into our concept. All kind of non-computer based board games like pictionary, any card game are good candidates but also many of the existing multi-user console games can gain additional benefit of the augmented reality setup. We think that inspired by our setup, game developers can find entirely new gaming ideas in 3D. However for the evaluation of our system we selected a traditional game which is widely known around the world and relies also on the use of social communication channels: Mah-Jongg.

The Game

Mah-Jongg is a very old traditional Chinese game, with roots going back into the dust of ancient centuries (Figure). In the beginning it's been only played by emperors and those in the know, because it's secret. Later public got access to it and in our century it become popular around the world. The high number of enthusiastic players developed many variants of the game, yet the basic rules are still kept.

We have selected Mah-Jongg as the major implementation example of our concept, because it is known worldwide, the main rules are easy to understand, and many of our system features can be presented with it. Mah-Jongg is played in our setup by sitting around the table wearing see- through HMDs. Each user also has a Personal Interaction Panel to manipulate the augmented game. The panel carries one player's tiles, which are manipulated with the stylus (Figure 76).

Figure 76. *Mah-Jongg*. The augmentation as seen from a third participant through the HMD.

Spatial Controlled Semantics

Interaction in our gaming applications covers besides some obvious system controls mostly direct object manipulation. Users in our environment mainly play the game by manipulating virtual objects (tiles, dices, and cards) in front of them. Most of the actions that will occur during direct manipulation are actions such as "Put-That-There" inspired by the motto of [Bolt, 1980]. Our concept relies on a proximity based drag-and-drop control model for 3D-user interaction. Such actions have in addition to the geometric transformation also semantic meaning.

Tiles are being transported from the wall to the user's hand, within the user's hand from one place to another to form groups, and from the hand to the table. We have defined several regions for the game: the wall region, the table region, the region on the PIP representing the user's hand, the region of the pen for the transition of tiles. In contrast to all other regions the pen region has no assigned snap condition geometry. The region transition of tiles into and from the region is triggered by pressing and releasing the button on the pen (see also Figure 38).

Players can independently manipulate their private tiles. Tiles snap to each other providing visual feedback and to trigger semantics. In this way groups can be formed and braked dynamically by pushing tiles together or moving them apart as shown in Figure 77. Grouping generates additional regions, allowing to add other tiles to that sub-region. Additionally the grouping mechanism gives feedback to the game engine to provide help information and determine game state.

Spatial controlled semantics read out from geometric actions has been verified as a powerful technique for game-piece manipulation. Using this paradigm, grouping of tiles and thus playing games is very natural and provides a high quality input to the game-engine, running the game logic.

Figure 77. A grouping sequence. A tile is grabbed with the pen and snapped to another tile to form a group. This action triggers semantics to insert the help-shovel showing additional help information. In this case the name of the combination is displayed.

Privacy in Mah-Jongg

In the design step we identified the logical regions of the common table and player owned regions to carry security information. In Mah-Jongg presets for one player are set, so that he or she can see and manipulate the own game-tiles, but no other player can see any valuable information. While a player moves one tile of his own set of tiles to the common tabletop, a region transition is triggered by the geometric constraints of snapping. This region-transition from one region to another causes the game engine to move the sub-graph of the game-tile from one place in the hierarchy into the place of the destination region, i.e. the tabletop-region. As the game-piece now inherits a new security information, it behaves differently, and is rendered visible for every player, due to the security settings of the common tabletop-region. Note, that the game engine does know nothing explicitly about security management, as the only step it makes, is to move one sub-hierarchy in the scene-graph into a different place.

Naturally the real game relies very much on the honesty of players not to look into other players tiles. This gives the game a secret and mystic touch. In general the rights to see tiles and to manipulate them is governed by game rules and tradition. The gaming situation decides what tiles can be picked up by which user. In the real game these conflict situations are solved over the social channel.

Our layering concept supports this type of privacy by assigning different security levels to regions. Tiles on a player-panel can only be seen and manipulated by him as shown in Figure 78. The texture containing the tile's sign is switched on and off by inheriting privacy information from the PIP region.

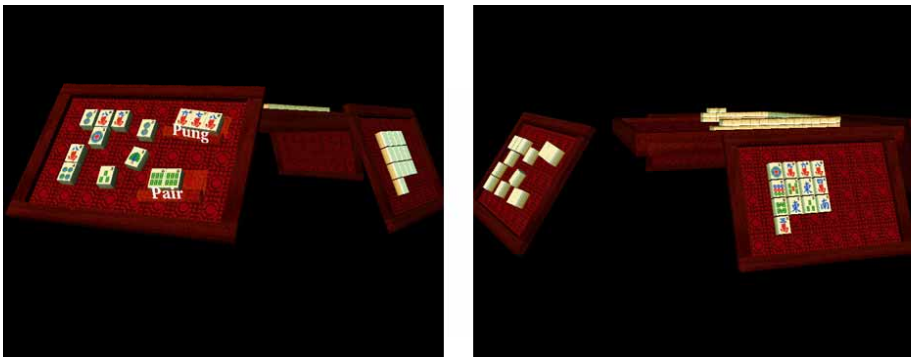


Figure 78. Privacy: The same situation during the game seen by two different players. Characters on the opponent's tiles, as well as help information is not shown. To keep a consistent augmentation, tiles are not hidden entirely from other players.

Picking up the tile by pointing with the pen inside it and pressing the button transfers the sub-graph of the tile to the pen region. The pen region has the same privacy settings like the PIP, so that other players still can't see the texture while manipulating the tile. Although the pen has the same security as the panel - it uses a reference to the PIP security settings -, it was necessary to define this region. The hierarchy of transformations allows thus to define a local pen coordinate system, which is transformed by tracker updates in world coordinates.

Moving the tile close enough to the table region, the tile snaps onto the surface, indicating a region transition. Releasing the pen button confirms this transition. As the security definition of the table region enables viewing for each user, the tile texture becomes visible and every player can see it as shown in Figure 79.

Although information layering supports privacy, the PIP incorporates simply by its physical properties an additional kind of privacy. Players may hold the panel in a position, so those tiles are visually hidden to other players, however they do not contain any useful information in all other players customized views.

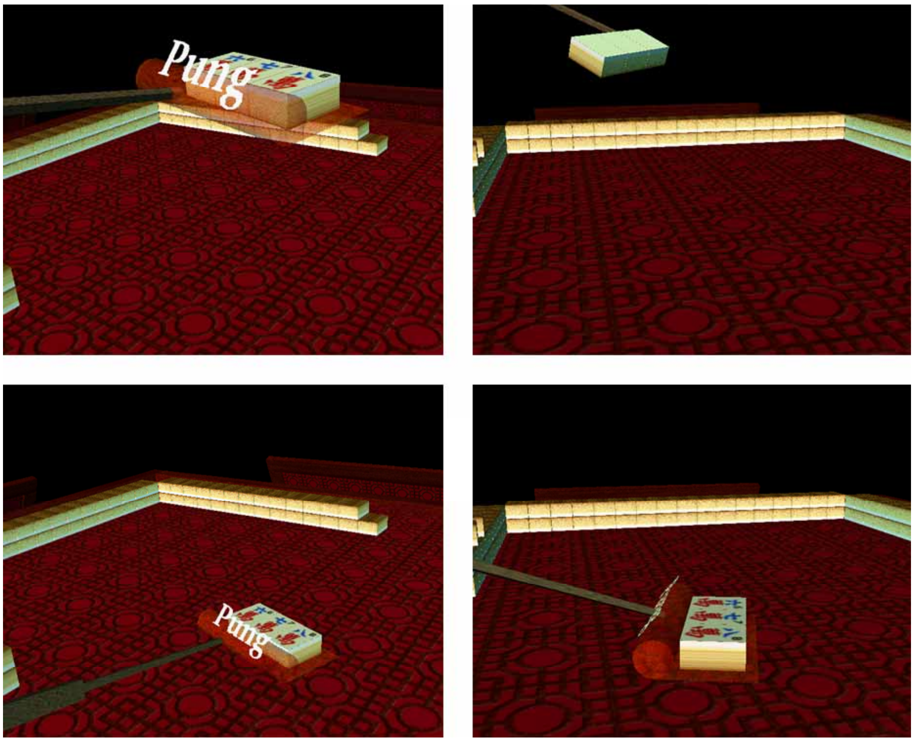


Figure 79. A player is placing a combination called *pung* onto the table (left column). A different player is observing this action (right column). This sequence gives an example for a region transition with changing privacy.

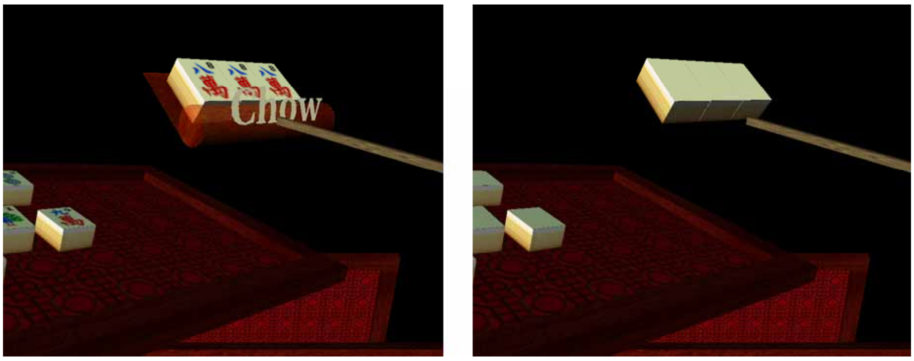


Figure 80. Private help expressed as a *shovel*. This additional handle helps to manipulate groups together. The same situation with another player's rights is shown on the right side.

Private Help

Additional information is provided to players on private layers (Figure 80). Our Mah-Jongg implementation has a semantic controlled help system supporting visual feedback for understanding, and additional geometry for group manipulations. Forming groups of tile results in the creation of a *shovel* placed under the grouped tiles. Text on the handle of the shovel indicates valid combinations. This kind of help-information can be used for more than status reporting. The owner of the group can also move the whole group by picking the handle of the shovel and moving it to another region. At the end of the game, shovels get displayed on the table to every user for easier calculation of scores.

Results

Basically our tests show, that it makes fun to play in such an environment! Even very inexperienced users find out very fast how the game is to be played without any additional information on the general use of the system. Tools like the described snapping mechanism can adequately compensate the shortcomings in precision of the hardware. Especially in regions where there are only a few snapping faces, the snapping conditions can be very generous, so that objects snap onto each other even if they are not very close. This allows rapid actions to be performed giving the game a dynamic character.

If object manipulation is restricted to drag-and-drop like actions, as presented in our examples, snapping is a powerful substitute for collision detection in virtual environments. As objects snap to each other when they are near, interpenetration happens very seldom and only in cases which do not disturb the user. Moreover, in some cases collision detection would rather hinder easy manipulation of objects.

Compared to the original game, the augmented version is capable of changing the whole mood of the game with additional 3D graphics. A simple yet effective demonstration of this concept is the application of *augmented hats* in Figure 76. Furthermore the flexible setup supports different types of games and the private help system is an additional feature.

Currently our system consists of a commercial available standard hardware, but we see a good chance that the system could be produced as a console game for multiple users in future. A *game-box* could contain game-server and rendering clients as well as the tracker source. If the game-box is placed on a table, players can sit around that table holding their pen and panel. The game-board is augmented on that table which results in a haptic feedback when placing tiles on the common playground.

As our hardware-setup is lent from a scientific-visualization system it is only natural to project our interaction techniques and privacy concepts back to that application area. We think that an adapted version of our system-concepts could enhance scientific visualization applications. Multiple scientists discussing common visualizations are able to switch on and off individualized information they like to see personally. Simplifications induced by the gaming domain could be removed to support other type of applications. We see a great potential for our setup to be used also for 3D education- and presentations-systems in the near future.

P. Picasso – Portait of Painter, after El Greco (1950)

Chapter Six - Translocation

A major result of our experiments to explore the general character of the presented Personal Interaction Panel though a series of applications under completely different circumstances encouraged following research efforts. Especially motivated by our observations in the gaming domain we abstracted the PIP interface concept to explore its suitability in other existing augmented and virtual environments. This section gives an overview of this strive.

6.1 PIP in Desktop VR Applications

The first attempt to transfer the underlying metaphor of the Personal Interaction Panel was to build an interface for a monitoring and control station prototype. The basic requirement dictated to construct a visualization system for a complex real-time database with over 2,000 different significant states. The proposed solution should exhibit the database for continuous inspection of the whole system state, but also visualize data of participating items in detail. A solution had to be presented for the navigation and interactive exploration of this database. Our group was selected by the European Space Agency (ESA) for the implementation of this project from 6 European competitors. Using this prototype and other demo systems ESA will evaluate the future use of VR technology in spacecraft operation. The project is done in a joint university-industry cooperation with the Belgian branch of Logica SA/SN company.

6.1.1 VR-MCS - a Spacecraft Monitoring and Control Environment

The objective of the *Virtual Reality Monitoring and Control System (VR-MCS)* project [Imagination, 1999] is to demonstrate the suitability of VR technology and techniques to the domain of spacecraft operations. The spacecraft monitoring and control environment to be developed in this project should enable a group of users – the ground operation team – to interact jointly with a single, shared environment representing the logical model of the on-board spacecraft system. Using a simulation of the European Module (APM) of the International Space Station (ISS) selected operation scenarios, like routine operations or operations during failures and anomalies can be tested in advance, training crew and ground control. Once installed in orbit, the system can work with real telemetry data and send back telecommands to the end-items in the module.

Graphical Structure

A complete analysis of the physical layout of the APM including all relevant mechanical structures, end-items, and circuits was carried out to develop a logical hierarchy of all elements. End-items were classified with respect to logical subsystems or functional groups. A schematic 2D and 3D representation for each end-item was designed. The 2D logical graph provides a hierarchy of functionality and functional grouping and is fixed screen-aligned in the plane of zero parallax. The placement of items directly in 3D on the other hand enables the representation of the real physical structure and leads to an improved understanding of spatial relationships and thus functional constraints. The graphical representation is enriched with the visualization of all related telemetry data describing status and measurements read in real time from life data streams.

Connections display flows in different circuits (e.g. water, air, data buses) revealing the internal status of the spacecraft. Figure 81 shows a screenshot of the system while investigating the Data Management Subsystem (DMS).

Personal Workspace

To interact with the system we designed a desktop-based display and interaction metaphor called *Personal Workspace* for the ground operation team (Figure 82). The desktop environment for each client supports stereo 3D output using shutter glasses. The two handed interface is derived from the Personal Interaction Panel metaphor and is carried out using custom 3D input devices. The setup consists of a pen in the dominant hand as a general pointing device and a prop – representing the data space – held with the non-dominant hand for simple rotations, translations and zooming operations. Both devices are tracked in position and orientation with *Ascension MiniBird* tracker and had buttons for mode switching (Figure 83).

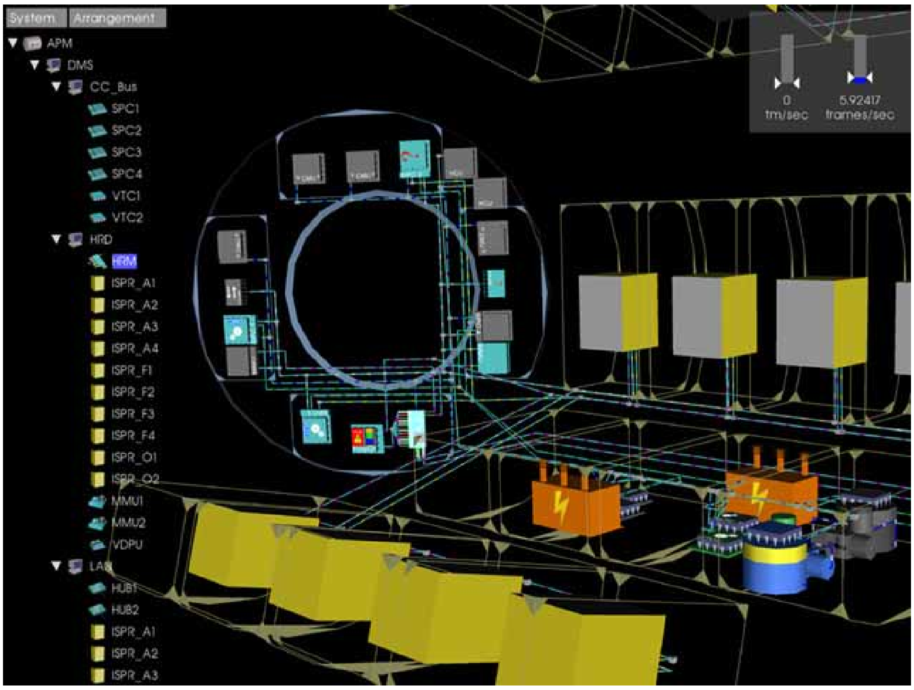


Figure 81. *VR-MCS* – Screenshot of the system while monitoring the operation of participating items.

Figure 82. *Personal Workspace* – System setup showing components of the desktop environment. Navigation prop and the pen of the *Wacom* tablet are tracked using a *MiniBird* system.

Figure 83. *Personal Workspace* – The actual hardware setup.

While the non-dominant hand is employed to orient and position the graphical structure of the database defining thus a frame of reference, the dominant hand performs fine grained and precise interaction in this frame. Furthermore a static, pressure sensitive (*Wacom*) sketchpad can be used for the common 2D graphical user interface elements like popup menus, etc. Thus the pen supports both direct manipulation of graphical elements in 3D and the fine-grained operation of desktop elements in a 2D metaphor. However in this arrangement the panel does not support two-handed interaction in the style of the PIP, the use of prop and pen do outweigh this drawback.

The general acceptance of the whole system was convincing; especially representatives of the training department liked the educative character of the representation. Both the visualization as well as the interaction metaphor was easy to understand. Users accepted the employed navigation metaphor (grabbing the information space by pushing the button on the prop) very quickly, however a little practice was needed for controlling the device.

We noticed a difference in interaction between Personal Workspace and the PIP interface. The two-handed character of the operation is only kept as long as the user pushes the button on the prop in the non-dominant hand. Is this button released, the bimanual frame of reference does not exist anymore, operation becomes a rather difficult absolute manipulation task in three-dimensional space. In addition this is unfortunate influenced by the accuracy of the tracking system that has to operate under the worst possible conditions – in front of an electromagnetic source and over the panel, which is a metallic device and causes distortions.

Analyzing the results we can summarize that it is possible to transfer the PIP interface into a desktop environment like also presented in [Sachs, 1991] and [Billinghurst, 1997], however no significant modifications should be introduced. A more careful investigation of the interaction scenario can improve performance in next generations of this type of interface.

Nevertheless the mixture of 3D and 2D interaction seems to be an interesting field for future investigations, as the navigation prop in the non-dominant hand can still be used when the pen is employed in combination with the pressure sensitive panel as a purely 2D interaction device. The overall system concept receives acknowledgment in other application fields, e.g. building infrastructure monitoring or the monitoring and control of complex machines on earth.

6.2 PIP for the CAVE Environment

While investigating the behavior of the interface in the Maze Game, described in chapter 5.5.2, we observed that the indirection of the physical manipulation and the visual feedback does not prevent users to interact with the PIP interface. The observation with the game inspired to try the interface in front of a projection, where similar to the desktop setup, the visible image is generated behind the place of actual interaction. The setup we use is a large screen projection where the user stands in front of the projection wall. Despite this difference in the output technology the employed hardware is similar to the augmented version of PIP.

As the user was not head-tracked in this environment we had to modify our application. We placed a virtual camera behind the users position that looked towards the environment. The experiment showed that it is possible to use the PIP in this environment, where only the virtual camera was registered in relation to the interaction props, but not the rest of the environment. In this type of VR setup navigation becomes much more important, as the virtual camera is not attached to the own viewpoint.

The CAVE system in [Cruz-Neira, 1993a] is basically comparable to our setup of the one-wall projection. It provides due to its arrangement a better coverage of the human visual field and uses time-interleaved stereoscopic projection that can be viewed with shutter glasses. Currently most CAVE applications use the *wand* input device to control applications. Based on our experience we think that the PIP interface could also support applications in the CAVE, where the leading user is even head-tracked for correct stereoscopy.

Despite being able to interact with the involved indirection similar to current state-of-the-art mouse based desktop systems, some test persons noted that they liked to see the interface directly on the panel. This inspired Schmalstieg [Schmalstieg, 1999a] to implement a transparent version of the Personal Interaction Panel.

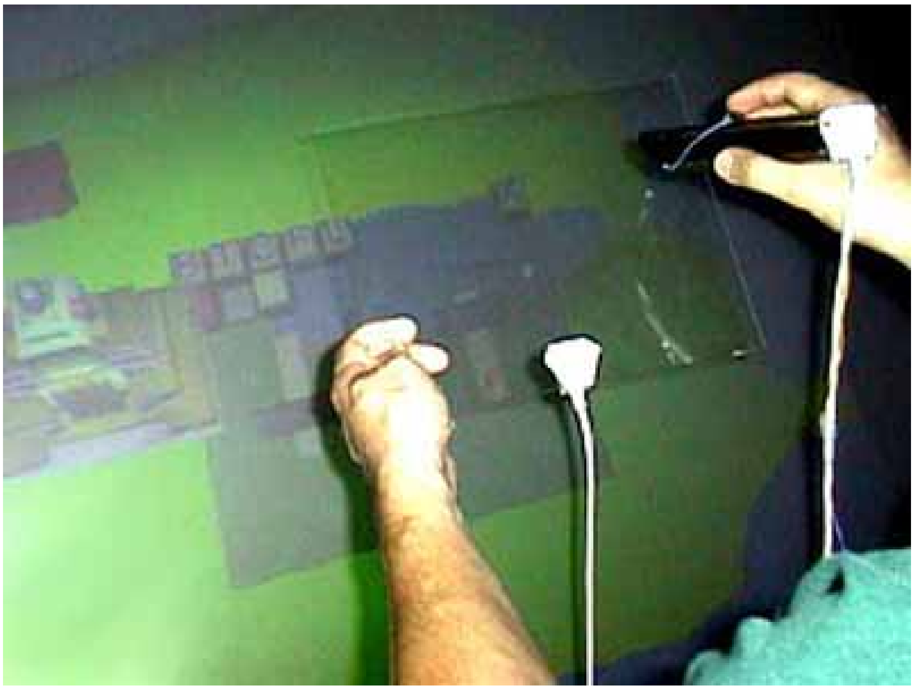


Figure 84. Using transparent props for interaction with the Virtual Table

6.3 PIP for Table Environments

The presented system in Figure 84 uses transparent props for two-handed interaction on the Barco BARON [Barco, 1997] Virtual Table (VT), a tabletop VR display based on a workbench metaphor [Krüger, 1995]. Users must wear shutter glasses to see the time-interleaved stereoscopic image. The head of the user interacting with the application using a modified PIP interface is tracked in position and orientation. The original pen and panel are replaced by transparent versions of the props. Based on the tracker information the system overlays the physical props using the back-projected display of the VT. This kind of inverse augmented reality we call *augmented VR*. The VT thereby provides an enhanced workspace with capable multipurpose tools. Following list summarized the implemented tools:

- ◊ *Tool and object palette*: The pad can carry tools, controls and offer collections of 3D objects to choose from as other PIP applications presented above.

- ◊ *Window tools*: Due to the transparent rendered virtual representation of the PIP the user can see through the panel into the scene, it becomes a see-through tool (as e.g. in [Wloka, 1995]).

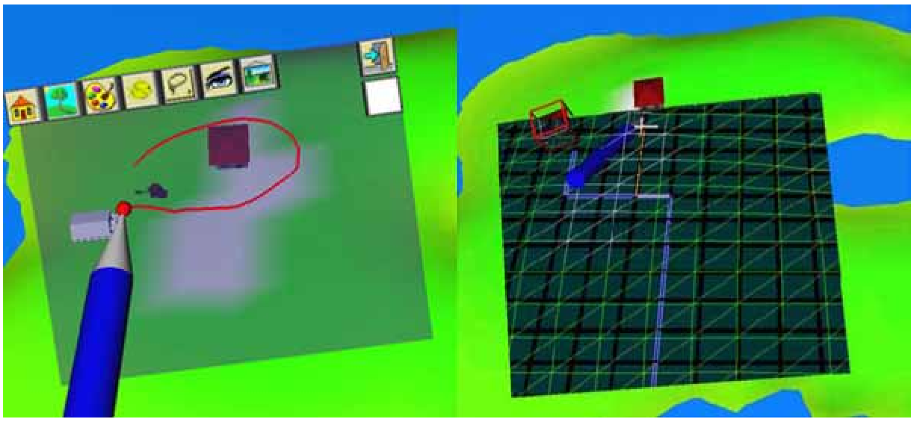


Figure 85. The *Virtual Landscaping* application allows the design of a city layout using sophisticated tools like the *through-the-plane selection* and the *X-ray* lens for underground manipulation.

- ◊ *Through-the-plane tool*: The user can orient the *window* defined by the pad and then manipulate objects as seen through the pad, i. e. manipulate the 2D projections of objects on the pad. This approach is similar to the image-plane interaction techniques as presented in [Pierce, 1997], however it has two important advantages over it: The 2D plane onto which objects are projected is easily manipulated by moving or orienting the pad without the need to move one's point-of-view. The physical surface of the pad provides a clear definition of the 2D manipulation plane and adds tactile feedback to operations performed on the 2D image.

- ◊ *Volumetric manipulation tool*: The pad itself can be used for active one-handed object manipulation (as e.g. in [Stoakley, 1995]). Exploiting the fact that the pad has a spatial extent, unlike the hot spot represented by the pen tip that has a theoretical dimension of zero. By sweeping the panel through space a volume is defined that can be used e.g. for selection of a number of objects, like the described *fish net* metaphor in [Schmalstieg, 1999a].

The metaphors are implemented by Schmalstieg into a *Virtual Landscaping* application that allows town planning in an augmented VR setup (Figure 85). Buildings and other terrain elements can be placed throughout the map representing the landscape. The landscape is projected in 3D on the table, giving the feeling that the placed objects reach out of the terrain.

A *cable TV rooting tool* allows the design of underground cables. To see under the ground an *X-ray lens* is used that is bound to the backside of the panel, thus immediately available whenever needed. *Information overlays* enrich the visualization and help to understand the presented data. *Through-the-plane selection* can be used to make quick selections from a large set of 3D data items. Finally a *3D history snapshot* function fixates different design stages in 3D, that can be reactivated if desired. Snapshots can be placed in free space, supporting comparison of design variants.

The system was informally tested with several users, most of which had computer (desktop) experience but little experience with VR systems. They generally found the design very appealing and were able to perform simple landscaping tasks after a few minutes of initial instruction. We did not observe any difficulties in understanding the tools. Complaints mainly addressed technical inadequacies like tracker error, lag or frame rate.

6.4 Proposed Further Application Areas and Environments

Using the basic underlying metaphor of the Personal Interaction Panel we have constructed many solutions for different environments. Relying on the experience learned from the process of design, introduction, and analysis in different scenarios, we can speculate to propose further *application fields* and *environments* of the metaphor.

From point of view of the *application field*, problems that require a direct interaction with 3D data can benefit from the interface. *Architectural design* or *town planning* for example need to draft layouts, modify, and perfect them to the final solution. Using the PIP interface this process can be enriched with the direct 3D character of the interaction and display. When integrated tightly in the whole design process, design drafts made using the direct manipulation techniques can serve as fundament for further processing. Similar engineering tasks could use the PIP for interaction with *virtual prototypes* of their concepts. In this case engineers can focus on the problem of prototype design, because the flexibility of the interface supports easily different variants of the design-model.

Similar to scientific visualization *Information Visualization* is becoming an important field of research in the last years. Complex multi-dimensional databases can be investigated using information visualization techniques to get more insight in the flood of numbers. Using 3D visualization methods for this data, navigation in the database and selection of or interaction with the elements becomes a crucial element of success. The PIP interface can support this task due to its naturalness and unencumbering character that was showed in different existing applications. Especially interactive information filtering techniques in combination with the SEAM mechanism [Schmalstieg, 1999b] seem to be promising in this field.

From the *application environment* perspective, the Personal Interaction Panel metaphor is closely related to the rapidly growing number of Personal Digital Assistants (PDA) as shown in Figure 20. Currently these devices are far from the flexibility and technology needed to implement a mobile version of the PIP, however they have the potential to develop in this direction. A basic criterion for the implementation is a solution for spatial awareness and registration with the real environment. Feiner shows in [Feiner, 1997] first steps towards this solution using differential GPS technology. As holographic technologies in free space are far from realization, in a first attempt autostereoscopic display technology on the panel, similar to the prototype presented in [D4D, 1999] could support 3D display without glasses. Finally the interaction device (e.g. pen) for manipulating the augmented content must also be tracked to enable handling of the information.

Finally we proposed internally a completely different approach for the PDA goal – to have information available anytime and everywhere. *Ubiquitous computing* research gathers around these ideas to show directions for future applications. Our proposal describes a future lightweight head-worn stereoscopic display, not much voluminous than today's sunglasses. The glasses accommodate miniature cameras for optical tracking and registration purposes. Whenever needed – e.g. during a discussion sitting around a table – the system searches for a flat surface in reach of the user (e.g. a sheet of empty paper on the tabletop) and augments a 3D display above the surface. If the user has some kind of pen style device in his hand, that is registered and serves further as interaction device to manipulate the augmented overlay. Using wireless dynamic networking technology sharing of the augmented information could be supported, thus making working with virtual 3D information an everyday task. This *vision* is of course a hard nut to crack if someone wants to implement it in every detail, however it can be defined of a far future goal to influence upcoming research in this field.

Zs. Szalavári – The Personal Interaction Panel
(1999)

Chapter Seven - Conclusions

7.1 *Concluding Remarks*

The way we perceive the continuously growing information environment will change dramatically over the next years. To achieve a solution for this problem *Augmented Reality* enhances human perception to experience the information space directly. This new technology intermixes the information environment with our real surroundings in three dimensions at interactive rates, providing a smooth immersion of the user by the coexistence of both spaces.

To interact with the perceived virtual information, research and development has to present new interaction metaphors that allow this seamless integration of the information space in our real world. A wide spectrum of devices and metaphors were presented in last years that are either specialized for a certain task or require the user to learn a special skill. An analysis of these interfaces led to the formulation of the *basic design guidelines* for interaction in augmented reality.

This work contributes to the problem of interacting in augmented reality with the introduction of a two-handed interface concept – the *Personal Interaction Panel (PIP)*.

The careful conception of the PIP interface resulted the insight that simple metaphors abstracted from everyday human skills for the interaction in a new environment help to increase the acceptance of the new interface. The less obtrusive a new technology is, the more willing are people to use it.

Resulting also from the investigations in the gaming domain, an important outcome is also the understanding how deep these skills are anchored in humans. The PIP interface conforms previous results in bimanual action, especially the ability for accomplishing manual tasks with a better performance when division of labor is asymmetric. Breaking this principle immediately can destroy the advantage of the proposed new device. Thus successful solutions on the way towards an intermixed everyday reality should carefully consider the human factor involved.

Furthermore, the analysis of the basic problem and other solutions helped to avoid the trap of being blinded by the fantastic features of a new technology. Indeed Augmented Reality certainly offers much more capabilities than today used and imagined, but again the human link in the chain should not be ignored. The presented interface reflects this insight in the incorporation of a 2D surface as the place of interaction in a three dimensional environment. Mankind is surrounded from the beginning of evolution with flat surfaces in the real world and this fact should not be omitted when designing a new interface.

The successful translations of the basic underlying metaphor to other Augmented- and Virtual Environments and the hard-hearted gaming domain give the opportunity to conclude, that the principle idea of the Personal Interaction Panel unifies many other earlier metaphors and makes it to a universal interface paradigm to interact with virtual content.

7.2 Considerations for the Future

Certainly it would be exciting to see even more applications using the proposed PIP interface. Both, modification of existing applications, as well as new applications could build their interface concept upon this work, profiting from the presented results and experiences.

This work can also be a fundamental topic of carrying out empirical measurements and an extended psychological study on the impacts of this interface. Results of such a work could possibly reflected back to the psychological research domain on bimanual action.

Finally ongoing cooperation with the Graz University of Technology provides promising results using optical tracking for determining position and orientation of the devices involved. A break-through in this field could make our interaction concept and interface wireless and *ubiquitous*, to be really seamlessly integrated in our everyday environment.

Chapter Eight - References

[Agrawala, 1995] Agrawala, M., Beers, A. C. Levoy, M.: 3D Painting on Scanned Surfaces. *Proceedings of 1995 ACM Symposium on Interactive 3D Graphics (I3DG'95)*, 1995, pp. 145-150.

[Agrawala, 1997] Agrawala, M., Beers, A. C., Fröhlich, B., Hanrahan, P.: The Two-User Responsive Workbench: Support for Collaboration Trough Individual Views of a Shared Space. *Computer Graphics (Proceedings of ACM SIGGRAPH'97 Conference)*, Vol. 31, August 1997, pp. 327-332.

[Amselem, 1995] Amselem, D.: A Window on Shared Virtual Environments. *Presence– Teleoperators and Virtual Environments*, Vol 4., No. 2, MIT Press, 1995, pp. 130-145.

[Angus, 1995] Angus, I. G., Sowizral, H. A.: Embedding the 2D Interaction Metaphor in a Real 3D Virtual Environment. *Proceedings of SPIE 2409 - Stereoscopic Displays and Virtual Reality Systems II*, Vol. 2409, 1995, pp. 282-293.

[Auer, 1998] Auer, T., Pinz, A., Gervautz, M.: Tracking in a Multi-User Augmented Reality System. *Proceedings of the IASTED International Conference Computer Graphics and Imaging (CGIM'98)*, M.H. Hamza (ed.), Halifax, Nova Scotia, Canada, June 1-4, 1998, pp. 249-252.

[Auer, 1999] Auer, T., Brantner, S., Pinz, A.: The Integration of Optical and Magnetic Tracking for Multi-User Augmented Reality. *Proceedings of 5th EUROGRAPHICS Workshop on Virtual Environments EGVE'99*, M. Gervautz, A. Hildebrand, D. Schmalstieg (eds.), Vienna, Austria, May 31-June 1, 1999, pp. 43 - 52.

[AVS, 1992] AVS: AVS Developers Guide - Release 4. *Advanced Visualization Systems Inc.*, 1992.

[Azuma, 1997] Azuma, R. T.: A Survey of Augmented Reality, *Presence–Teleoperators and Virtual Environments*, Vol. 6, No. 4, MIT Press, 1997, pp. 355-385.

[Bajura, 1992] Bajura, M., Fuchs, H., Ohbuchi, R.: Merging Virtual Objects with the Real World: Seeing Ultrasound Imaginery within the Patient. *Computer Graphics (Proceedings of ACM SIGGRAPH'92 Conference)*, Vol. 26, No. 2, August, 1992, pp. 203-210.

[Bajura, 1995] Bajura, M., Neumann, U.: Dynamic Registration Correction in Augmented-Reality Systems. *Proceedings of the Virtual Reality Annual Symposium VRAIS'95*, 1995, pp. 189-197.

[Barco, 1997] Barco Projection Systems: Barco BARON Table. *Barco Projection Systems Web Information*, *http://www.barco.com/projecti/products/bsp/baron.htm*, 1997.

[Bier, 1990] Bier, E.: Snap-Dragging in three Dimensions. *Proceedings of 1990 ACM Symposium on Interactive 3D Graphics (I3DG'90)*, Vol. 24, No. 2, March 1990, pp. 193-204.

[Bier, 1993] Bier, E., Stone, M., Pier, K., Buxton, W., DeRose, T.: Toolglass and Magic Lenses: The See-through Interface. *Computer Graphics (Proceedings of ACM SIGGRAPH'93 Conference)*, Vol. 27, No. 2, August 1993, pp. 73-80.

[Billinghurst, 1996] Billinghurst, M., Weghorst, S., Furness, T. III: Shared Space: An Augmented Reality Interface for Computer Supported Collaborative Work. *Proceedings of Collaborative Virtual Environments'96*, 1996.

[Billinghurst, 1997] Billinghurst, M., Baldis, S., Matheson, L., Philips, M.: 3D Palette: A Virtual Reality Content Creation Tool. *Proceedings of 1997 ACM Symposium on Virtual Reality Software and Technology (VRST'97)*, 1997, pp. 155-156.

[Bolt, 1980] Bolt, R.A.: "Put-That-There": Voice and Gesture at the Graphics Interface. *Computer Graphics (Proceedings of ACM SIGGRAPH '80 Conference)*, Vol. 14, No. 3, July, 1980, pp. 262-270.

[Broll, 1995] Broll, W.: VRML: From the Web to Interactive Multi-User Virtual Reality. *Proceedings of GI Workshop on Modelling, Virtual Worlds, and Distributed Graphics*, Bonn, 1995.

[Brooks, 1990] Brooks, F. P., Jr., Ouh-Young, M., Batter, J. J., Kilpatrick, P. J.: Project GROPE - Haptic Displays for Scientific Visualization. *Computer Graphics (Proceedings of ACM SIGGRAPH'90 Conference)*, Vol. 24, No. 4, August 1990, pp. 177-185.

[Brooks, 1996] Brooks, F. P., Jr.: The Computer Scientist as Toolsmith II. *Communications of the ACM*, Vol. 39, No. 3, 1996, pp. 61-68.

[Bryson, 1991] Bryson, S.: The Virtual Wind Tunnel. *Proceedings of IEEE Visualization'91*, 1991, pp.17-25.

[Bryson, 1993] Bryson, S.: The Distributed Virtual Wind Tunnel. *Proceedings of Supercomputing '92* also in *ACM SIGGRAPH'93 Course Notes, No. 43*, 1993, pp: 3.1-3.10.

[Bukowski, 1995] Bukowski, R. W., Séquin, C. H.: Object Associations - A Simple and Practical Approach to Virtual 3D Manipulation. *Proceedings of 1995 ACM Symposium on Interactive 3D Graphics (I3DG'95)*, Vol. 28, April 1995, pp. 131-138.

[Butterworth, 1992] Butterworth, J., Davidson, A., Hench, S, Olano, M.: 3DM - A Three Dimensional Modeler Using a Head-Mounted Display. *Proceedings of 1992 ACM Symposium on Interactive 3D Graphics (I3DG'92)*, Vol. 25, No. 2, 1992, pp. 131-138.

[Buxton, 1986] Buxton, W., Myers, B.: A Study in Two-Handed Input. *Proceedings of Computer Human Interaction CHI'86*, 1986, pp. 321-326.

[Buxton, 1990] Buxton, W.: There's more to Interaction than Meets the Eye: Some issues in manual input. In J. Preece, L. Keller and H. Stolk, editors, *Human-Computer Interaction*. Prentice-Hall. 1990, pp. 122-137.

[Carlsson, 1993] Carlsson, C., Hagsand, O.: DIVE - A Platform for Multi-User Virtual Environments. *Computers & Graphics*, Vol. 17, No. 6., 1993, pp. 663-669.

[Chen, 1988] Chen, M., Mountford, S., Sellen, A: A Study in Interactive 3-D Rotation Using 2-D Control Devices. *Computer Graphics (Proceedings of ACM SIGGRAPH '88 Conference)*, Vol. 22, 1988, pp. 121-129.

[Cohen, 1995] Cohen, J. D., Lin, M. C., Machova, D., Ponamgi, M.: I_COLLIDE: An Interactive and Exact Collision Detection System for Large-Scale Environments, *Proceedings of 1995 ACM Symposium on Interactive 3D Graphics (I3DG'95)*, Vol. 28, April 1995, pp. 189-196.

[Conner, 1992] Conner, D.B., Snibbe, S.S., Herndon, K.P., Robbins, D.C., Zeleznik, R.C., Dam, A. van: Three-Dimensional Widgets. *Proceedings of 1992 ACM Symposium on Interactive 3D Graphics (I3DG'92)*, Vol. 25, No. 2, 1992, pp. 183-188.

[Cruz-Neira, 1993a] Cruz-Neira, C., Sandin, D.J., De-Fanti, T.A.: Surround Screen Projection-Based Virtual Reality: The Design and Implementation of the Cave, *Computer Graphics (Proceedings of ACM SIGGRAPH'93 Conference)*, Vol. 27, August 1993, pp. 135-142.

[Cruz-Neira, 1993b] Cruz-Neira. C., et. al.: Scientists in Wonderland: A Report on Visualization Applications in the CAVE Virtual Reality Environment. *Proceedings of the IEEE 1993 Symposium on Research Frontiers in Virtual Reality*, *Visualization '93*, San Jose, CA, USA, October 1993, pp. 59-66.

[Cutler, 1997] Cutler, L. D., Fröhlich, B., Hanrahan, P.: Two-Handed Direct Manipulation on the Responsive Workbench, *Proceedings of 1997 ACM Symposium on Interactive 3D Graphics (I3DG'97)*, Providence, Richmond, USA, 1997, pp. 107-114.

[D4D, 1999] Technical University of Dresden: D4D – The Dresden 3D Display, *http://kastor.inf.tu-dresden.de/D4D/*, 1999.

[Darell, 1994] Darell, T., Maes, P., Blumberg, B., Pentland, A. P.: A Novel Environment for Situated Vision and Behavior. *Available as Technical Report No. 261, MIT Media Laboratory, Perceptual Computing Section*, 1994.

[Darken, 1993] Darken, R., Sibert, J.: A Toolset for Navigation in Virtual Environments, *Proceedings of ACM User Interface Software and Technology UIST'93*, 1993, pp. 157-165.

[Edwards, 1993] Edwards, E. K., Rolland, J. P., Keller, K. P.: Video See-through Design for Merging of Real and Virtual Environments, *Proceedings of the Virtual Reality Annual Symposium VRAIS'93*, 1993, pp. 222-233.

[Edwards, 1997] Edwards, J., Hand, C.: MaPS: Movement and Planning Support for Navigation in an Immersive VRML Browser. *Proceedings of VRML'97*, 1997, pp. 67-75.

[Fa, 1993] Fa, M., Fernando, T., Dew, P.M.: Direct 3D Manipulation Techniques for Interactive Constraint-based Solid Modeling, *Computer Graphics Forum (Proceedings of EUROGRAPHICS'93)*, Vol. 12, No. 3, 1993, pp. 237-248.

[Fahlén, 1993] Fahlén, L.E., Brown, C.G., Ståhl, O. and Carlsson, C.: A Space Based Model for User Interaction in Shared Synthetic Environments. *Proceedings of INTERCHI'93*, 1993, pp. 43-48.

[Faisstnauer, 1998] Faisstnauer, C., Schmalstieg, D., Szalavári, Zs., Computer-Assisted Selection of 3D Interaction and Navigation Metaphors, *Proceedings of the 1998 VRST Workshop on Computer Graphics*, Taipei, Taiwan, Nov. 5-6, 1998, p. 4.

[FakeSpace, 1999a] FakeSpace Inc.: Immersive WorkBench. *FakeSpace Inc. Web Information*, http://www.fakespace.com/products/immersiv.html, 1999.

[FakeSpace, 1999b] FakeSpace Inc.: PINCH Gloves. *FakeSpace Inc. Web Information*, http://www.fakespace.com/products/pinch.html, 1999.

[Feiner, 1992] Feiner, S., MacIntyre, B. and Seligmann, D.: Annotating the Real World with Knowledge-Based Graphics on a See-Through Head-Mounted Display. *Proceedings of Graphics Interface'92*, 1992, pp. 78-85.

[Feiner, 1993] Feiner, S., MacIntyre, B. and Seligmann, D.: Knowledge-Based Augmented Reality. *Communications of the ACM*, Vol. 36, No. 7, pp. 53-62.

[Feiner, 1997] Feiner, S., MacIntyre, B., Höllerer, T., Webster, T.: A Touring Machine: Prototyping 3D mobile augmented reality systems for exploring the urban environment. *Proceedings of ISWC '97 (Internatinal Symposium on Wearable Computers)*, October 13-14, 1997, Cambridge, MA. Also as: *Personal Technologies*, Vol. 1, No. 4, 1997, pp. 208-217.

[Fritzmaurice, 1993] Fritzmaurice, G.W.: Situated Information Spaces and Spatially Aware Palmtop Computers. *Communications of the ACM*, Vol. 36, No. 7, 1993, pp. 39-49.

[Fuhrmann, 1997] Fuhrmann, A., Löffelmann, H., Schmalstieg, D.: Collaborative Augmented Reality: Exploring Dynamical Systems, *Proceedings of IEEE Visualization '97*, October 1997, pp. 459-462.

[Fuhrmann, 1998] Fuhrmann, A., Gröller, M. E.: Real-Time Techniques for 3D Flow Visualization. *Proceedings of IEEE Visualization '98*, November 1998, pp. 305-312.

[Fuhrmann, 1999] Fuhrmann, A., Schmalstieg, D.: Concept and Implementation of a Collaborative Workspace for Augmented Reality, *Available as Technical Report TR-186-2-99-04, Institute of Computer Graphics, Vienna University of Technology*, ftp://ftp.cg.tuwien.ac.at/pub/TR/99/TR-186-2-99-04Paper.pdf, 1999, p. 11.

[Gervautz, 1996] Gervautz, M., Schmalstieg, D., Szalavári, Zs., Karner, K., Madritsch, F., Pinz, A.: "Studierstube" - A Multi-User Augmented Reality Environment for Visualization and Education. *Available as Technical Report TR-186-2-96-10, Institute of Computer Graphics, Vienna University of Technology*, ftp://ftp.cg.tuwien.ac.at/pub/TR/96/TR-186-2-96-10Paper.ps.gz, 1996, p. 8.

[Glasstron, 1999] SONY Corp.: SONY Glasstron LDI-D100BE. *SONY Corp. Web Information*, http://www.sony-europe.com/cons/glasstron/ or http://bpgprod.sel.sony.com/nab/product/ldi.htm, 1999.

[Goble, 1995] Goble, J., Hinckley, K., Pausch, R., Snell, J., Kassel, N.: Two-Handed Spatial Interface Tools for Neurosurgical Planning. *IEEE Computer*, Vol. 28, No. 7, 1995, pp. 20-26.

[Gottschalk, 1996] Gottschalk, S., Lin, M.C., Manocha, D., OBB-Tree: A Hierarchical Structure for Rapid Interference Detection, *Computer Graphics (Proceedings of ACM SIGGRAPH '96 Conference)*, Vol. 30, 1996, pp. 171-180.

[Gröller, 1996] Gröller, E., Wegenkittl, R., Milik, A., Prskawetz, A., Feichtinger, G., Sanderson, W.C.: The Geometry of Wonderland. *Chaos, Solitons & Fractals*, Vol. 7., No. 12, 1996, pp. 1989-2006.

[Guiard, 1987] Guiard, Y.: Asymmetric Division of Labor in Human Skilled Bimanual Action: The Kinematic Chain as Model. *Journal of Motor Behaviour*, Vol. 19, No. 4, 1987, pp. 486-517.

[Hand, 1995] Hand, C.: A Survey of 3-D Input Devices. *Technical Report TR94/2*, Department of Computer Science, De Monfort University, UK, 1994.

[Hand, 1997] Hand, C.: A Survey of 3D Interaction Techniques. *Computer Graphics Forum*, Vol. 16, No. 5, 1997, pp. 269-281.

[Harmon, 1996] Harmon, R., Patterson, W., Ribarsky, W., Bolter, J.: The Virtual Annotation System. *Proceedings of the Virtual Reality Annual Symposium VRAIS '96*, 1996, pp. 239-245.

[Hartman, 1996] Hartman, J., Wernecke, J.: The VRML 2.0 Handbook: Building Moving Worlds on the Web. Addison-Wesley, New York, 1996.

[Hinckley, 1994a] Hinckley, K., Pausch, R., Goble, J.C., Kassell, N.F.: A Survey of Design Issues in Spatial Input. *Proceedings of ACM User Interface Software and Technology UIST'94*, 1994, pp. 213-222.

[Hinckley, 1994b] Hinckley, K., Pausch, R., Goble, J.C., Kassell, N.F.: Passive Real-World Interface Props for Neurosurgical Visualization. *Proceedings of ACM CHI '94 Conference on Human Factors in Computing Systems*, 1994, pp. 452-458.

[Hinckley, 1997a] Hinckley, K., Pausch, R., Proffitt, D.: Attention and Visual Feedback: The Bimanual Frame of Reference, , *In Proceedings of 1997 ACM Symposium on Interactive 3D Graphics (I3DG'97)*, Providence, RI, USA, 1997, pp. 121-126.

[Hinckley, 1997b] Hinckley, K., Pausch, R., Proffitt, D., Patten, J., Kassell, N.: Cooperative Bimanual Action. *Proceedings of ACM CHI '97 Conference on Human Factors in Computing Systems*, 1997, pp. 27-34.

[Holmgren, 1992] Holmgren, D.: Design and Construction of a 30-Degree See-Through Head-Mounted-Display. *Available as Technical Report 92-030 at the University of North Carolina*, ftp://ftp.cs.unc.edu/pub/technical-reports/92-030.ps.Z , 1992.

[Imagination, 1999] Imagination Computer Services GesmbH.: VR-MCS: Virtual Reality Monitoring and Control System, *Project Documentation*, 1999.

[Jacoby, 1993] Jacoby, R. H.: Using virtual menus in a virtual environment. *ACM SIGGRAPH'93 Course Notes*, No. 43, 1993, pp. 7.1 - 7.8.

[Juster, 1992] Juster, N.P.: Modeling and Representation of Dimensions and Tolerances: a Survey. *CAD*, Vol. 24, No. 1, January 1992, pp. 3-17.

[Kabbash, 1994] Kabbash, P., Buxton, W., Sellen, A.: Two-Handed Input in a Compound Task. *Proceedings of CHI '94 Conference on Human Factors in Computing Systems*, 1994, pp. 417-423.

[KEO, 1999] Kaiser Electro Optics Inc.: Head Up Displays, *KEO Web Information*, http://www.keo.com/Product_EO_HUDs.html, 1999.

[Krüger, 1995] Krüger, W., Bohn, C., Fröhlich, B., Schüth, H., Strauss, W., Wesche, G.: The Responsive Workbench: A Virtual Work Environment. *IEEE Computer*, Vol. 28, No. 7, July, 1995, pp. 42-48.

[Löffelmann, 1996] Löffelmann, H., Mroz, L., Gröller, E., Purgathofer, W.: Stream arrows: Enhancing the Use of Streamsurfaces for the Visualization of Dynamical Systems. *The Visual Computer*, Vol. 13, No. 8, 1997, pp. 359-369.

[Löffelmann, 1997] Löffelmann, H., Gröller, E.: DynSys3D: A workbench for developing advanced visualization techniques in the field of three-dimensional dynamical systems. *Proceedings of The Fifth International Conference in Central Europe on Computer Graphics and Visualization '97 (WSCG '97)*, University of West Bohemia, Plzen, Czech Republic, February, 1997, pp. 301-310.

[Mapes, 1995] Mapes, D. P., Moshell, J. M.: A Two-Handed Interface for Object Manipulation in Virtual Environments, *Presence– Teleoperators and Virtual Environments*, Vol. 4, No. 4, MIT Press, 1995, pp. 403-416.

[Mel, 1988] Mel, B. W., Omohundro, S. M., Robinson, A. D., Skiena, S. S., Thearling, K. H., Wolfram, S., Young, L. T.: Tablet: The Personal Computer in the Year 2000. *Communications of the ACM*, Vol. 31, No. 6, 1988, pp. 638-648.

[Mine, 1997] Mine, M.R., Brooks Jr., F. P., Sequin, C. H.: Moving Objects in Space: Exploiting Proprioception In Virtual-Environment Interaction, *Computer Graphics (Proceedings of ACM SIGGRAPH '97 Conference)*, Vol. 31, 1997, pp.19-34.

[Nielson, 1986] Nielson, G., Olsen, D.: Direct Manipulation Techniques for 3D Objects Using 2D Locator Devices, *Proceedings SIGGRAPH Workshop on Interactive 3D Graphics*, 1986, pp. 175-182.

[Nintendo, 1989] Nintendo Inc., Mattel Inc.: The Power Glove. *Chris Hand HomeBrew VR Web Information*, http://www.cms.dmu.ac.uk/~cph/pg.html, 1989.

[Pausch, 1995] Pausch, R., Burnette, T., Brockway, D., Weiblen, M.: Navigation and Locomotion in Virtual Worlds via Flight into Hand-Held Miniatures. *Computer Graphics (Proceedings of ACM SIGGRAPH '95 Conference)*, Vol. 29, 1995, pp. 399-401.

[Pausch, 1996] Pausch, R., Snoddy, J., Taylor, R., Watson, S., Haseltine, E.: Disney's Aladdin: First Steps Towards Storytelling in Virtual Reality. *Computer Graphics (Proceedings of ACM SIGGRAPH '96 Conference)*, Vol. 30, 1996, pp.193-203.

[Petta, 1999] Petta, P., Staller, A., Trappl, R., Mantler, S., Szalavári, Zs., Psik, T., Gervautz, M.: Towards Engaging Full-Body Interaction, *8th International Conference on Human-Computer Interaction (HCI International'99)*, August 22-27, Munich, Germany, 1999.

[Peuchot, 1995] Peuchot, B., Tanguy, A., Eude, M.: Virtual Reality as an Operative Tool During Scoliosis Surgery. *Proceedings of Computer Vision, Virtual Reality, and Robotics in Medicine '95*, 1995, pp. 549-554.

[Pierce, 1997] Pierce, J.S., Forsberg, A., Conway, M. J., Hong, S., Zeleznik, R., Mine, M.R.: Image Plane Interaction Techniques in 3D Immersive Environments. *Proceedings of 1997 ACM Symposium on Interactive 3D Graphics (I3DG'97)*, Providence, Richmond, USA, 1997, pp. 39-43.

[Poupyrev, 1998] Poupyrev, I., Tomokazu, N., Weghorst, S.: Virtual Notepad: Handwriting in Immersive VR. P*roceedings of the Virtual Reality Annual Symposium VRAIS'98*, 1998, pp.126-132.

[Pyramid, 1999] Pyramid Systems Inc.: Immersadesk. *Pyramid Systems Inc. Web Information*, http://www.pyramidsystems.com/idesk.html, 1999.

[Rekimoto, 1995] Rekimoto, J., Nagao, K.: The World through the Computer: Computer Augmented Interactions with Real World Environments, *Proceedings of ACM User Interface Software and Technology UIST '95*, 1995, pp. 29-36.

[Rekimoto, 1996] Rekimoto, J.: TransVision: A Hand-held Augmented Reality System for Collaborative Design, *Proceedings of Virtual Systems and Multimedia VSMM'96*, Gifu, Japan, 18-20 September, 1996, p. 6.

[Rose, 1995] Rose, E., Breen, D., Ahlers, K. H., Crampton, C., Tuceryan, M., Whitaker, R., Greer, D.: Annotating Real-World Objects Using Augmented Reality. *Computer Graphics Developments in Virtual Environments*, 1995, pp. 357-370.

[Sachs, 1991] Sachs, E., Roberts, A., Stoops, D.: 3-Draw: A Tool for Designing 3D Shapes. *IEEE Computer Graphics & Applications*, Vol. 11, No. 6, 1991, pp. 18-26.

[Schmalstieg, 1996] Schmalstieg, D., Szalavári, Zs., Fuhrmann, A., Gervautz, M.: "Studierstube"- An Environment for Collaboration in Augmented Reality. *Proceedings of 1996 Workshop on Collaborative Virtual Environments CVE'96*, Nottingham, UK, September 19-20, 1996, p.2.

[Schmalstieg, 1997] Schmalstieg, D., Gervautz, M.: Modeling and Rendering of Outdoor Scenes for Distributed Virtual Environments. *Proceedings of ACM Symposium on Virtual Reality Software and Technology 1997 (VRST'97)*, Lausanne, Switzerland, September 15-17, 1997, pp. 209-216.

[Schmalstieg, 1999a] Schmalstieg, D., Encarnação, M., Szalavári, Zs.: Using Transparent Props for Interaction with the Virtual Table, *Proceedings of 1999 ACM Symposium on Interactive 3D Graphics (I3DG'99)*, Atlanta, GA, USA, April 26-28, 1999, pp. 147-154.

[Schmalstieg, 1999b] Schmalstieg, D., Schaufler, G.: Sewing Virtual Worlds Together with SEAMS: A Mechanism to Construct Large Scale Virtual Environments. *Presence – Teleoperators and Virtual Environments*, Vol. 8, No. 4, MIT Press, 1999.

[Serra, 1995] Serra, L., Poston, T., Hern, N., Choon, C.B., Waterworth, J. A.: Interaction Techniques for a Virtual Workspace. *Proceedings of 1995 ACM Symposium on Virtual Reality Software and Technology (VRST'95)*, also as online document at http://ciemed.iss.nus.sg/research/3dinterfaces/papers/vrst95/vrst95-1.html, 1995, pp. 79-80.

[Sharma, 1996] Sharma, R., Molineros, J.: Interactive Visualization and Augmentation of Mechanical Assembly Sequences. *Proceedings of Graphics Interface '96*, 1996, pp. 230-237.

[Shaw, 1996] Shaw, C. D.: Two-Handed 3D Editing, *Position Paper for CHI'96 Workshop on Psychological Issues of VE Interfaces*, Vancouver, British Columbia, Canada, Apr. 14-15, 1996, p. 5.

[Shaw, 1998] Shaw, C. D.: Pain and Fatigue in Desktop VR: Initial Results, *Proceedings of Graphics Interface '98*, Vancouver, British Columbia, Canada, June 18-20, 1998, p. 185-192.

[Smith, 1997] Smith, G., Mariani, J.: Using Subjective Views to Enhance 3D Applications, *Proceedings of 1997 ACM Symposium on Virtual Reality Software and Technology (VRST'97)*, 1997, pp. 139-146.

[Sowizral, 1994] Sowizral, H. A.: Tools and Metatools for Interacting with a Virtual Environment, *Proceedings of SPIE 2177 - Stereoscopic Displays and Virtual Reality Systems*, Vol. 2177, San Jose, USA, February 8-10, 1994, pp. 409-416.

[State, 1996a] State, A., Hirota, G., Chen, D., Garrett, W., Livingston, M.: Superior Augmented Reality Tracking by Integrating Landmark Tracking and Magnetic Tracking. *Computer Graphics (Proceedings of ACM SIGGRAPH '96 Conference)*, Vol. 30, 1996, pp. 429-438.

[State, 1996b] State, A., Livingston, M.A., Garrett, F., Hirota, G., Whitton, M.C., Pisano, E.D., Fuchs, H.: Technologies for Augmented Reality Systems: Realizing Ultrasound-Guided Needle Biopsies. *Computer Graphics (Proceedings of ACM SIGGRAPH '96 Conference)*, Vol. 30, 1996, pp. 439-446.

[StbAPI, 1998] StbAPI: Studierstube Application Programmer Interface (StbAPI Release 1.1/1.2beta), *http://www.cg.tuwien.ac.at/groups/vr/studierstube*, 1998.

[Stoakley, 1995] Stoakley, R., Conway, M. J., Pausch, R.: Virtual Reality on a WIM: Interactive Worlds in Miniature. *Proceedings of CHI'95 Conference on Human Factors in Computing Systems*, 1995, pp. 265-272.

[Strauss, 1992] Strauss, P., Carey, R.: An Object Oriented 3D Graphics Toolkit. *Computer Graphics (Proceedings of ACM SIGGRAPH '92 Conference)*, Vol. 26, No. 2, 1992, pp. 341-347.

[Sturman, 1994] Sturman, D. J., Zeltzer, D.: A Survey of Glove-Based Input. *IEEE Computer Graphics and Applications*, January, 1994, pp. 30-39.

[Sutherland, 1968] Sutherland, I.: A Head-Mounted Three Dimensional Display. *Proceedings of AFIPS Fall Joint Computer Conference 33*, 1968, pp. 757-764.

[Szalavári, 1997] Szalavári, Zs., Gervautz, M.: The Personal Interaction Panel – A Two-Handed Interface for Augmented Reality, *Computer Graphics Forum (Proceedings of EUROGRAPHICS'97)* Vol. 16, No. 3, September, 1997, pp. 335-346.

[Szalavári, 1998a] Szalavári, Zs., Schmalstieg, D., Fuhrmann, A., Gervautz, M.: "Studierstube" – An Environment for Collaboration in Augmented Reality. *Journal of the Virtual Reality Society – Virtual Reality – Systems, Development and Applications*, Vol. 3, No. 1, 1998, pp. 37-49.

[Szalavári, 1998b] Szalavári, Zs., Eckstein, E., Gervautz, M.: Collaborative Gaming in Augmented Reality, *Proceedings of 1998 ACM Symposium on Virtual Reality Software and Technology (VRST'98)*, Taipeh, Taiwan, November 2-5, 1998, pp.195-204.

[Taylor, 1993] Taylor, R. M., Robinett, W. et. al.: The nanoManipulator: A Virtual Reality Interface for a Scanning Tunneling Microscope. *Computer Graphics (Proceedings of ACM SIGGRAPH '93 Conference)*, Vol. 27, pp. 127-134.

[Viega, 1996] Viega, J., Conway, M.J., Williams, G., Pausch, R..: 3D Magic Lenses, *Proceedings of ACM User Interface Software and Technology UIST'96*, 1996, pp. 51-58.

[VTI, 1999] Virtual Technologies Inc.: CyberGlove. *Virtual Technologies Web Information*, http://www.virtex.com/products/hw_products/cyberglove.html, 1999.

[Ware, 1997] Ware, C., Fleet, D.: Context Sensitive Flying Interface, *Proceedings of 1997 ACM Symposium on Interactive 3D Graphics (I3DG'97)*, Providence, RI, USA, 1997, pp. 127-130.

[Webster, 1996] Webster, A., Feiner, S., MacIntyre, B., Massie, W., Krueger, T.: Augmented Reality in Architectural Construction, Inspection and Renovation, *Proceedings of ASCE Third Congress on Computing in Civil Engineering,* Anaheim, CA, USA, June 17-19, 1996, pp. 913-919.

[Weiser, 1991] Weiser, M.: The Computer for the 21st Century. *Scientific American*, Vol. 265, No. 3, 1991, pp. 94-104.

[Wloka, 1995] Wloka, M. M., Greenfield, E.: The Virtual Tricorder: A Uniform Interface for Virtual Reality, *Proceedings of ACM User Interface Software and Technology UIST '95*, 1995, pp. 39-40.

[Zeleznik, 1997] Zeleznik, R. C., Forsberg, A. S., Strauss, P. S.: Two Pointer Input For 3D Interaction, *Proceedings of 1997 ACM Symposium on Interactive 3D Graphics (I3DG'97)*, Providence, RI, USA, 1997, pp. 115-120.

Chapter Nine - Appendices

9.1 Curriculum Vitae

SZALAVÁRI Zsolt Szabolcs

Personal information:

Date of Birth: 02/28/1972
Place of Birth: Budapest / Hungary
Mother: Szalavári Erzsébet
Father: Szalavári Ottó
Status: married on 10/21/1995 with VAJDA Veronika
daugther Zsófia Veronika born on 02/21/1999
Languages: Hungarian, German, English, and Russian

Education and Work Experience

Aug. 1995 -	PhD research assistant in the VR-Lab at the Institute of Computer Graphics, Vienna University of Technology
Jan. 1993 - Jun. 1995	Diploma from the Budapest University of Technology Faculty of Mechanical Engineering / Process Design, Department of Fluid Dynamics
Feb. 1995 - May 1995	Diploma Thesis on “Raytracing of natural, atmospheric light phenomena”, Institute of Computer Graphics, Vienna University of Technology
Sep.1994	Summer Course in High Power Lasers and Applications, Certificate, Cluny-Paris/France
Sep. 1993 - Jan. 1994	Scholarship at the Vienna University of Technology
Aug. 1993	Summer Course in Holography, Lund/Sweden Certificate and Diploma in Holography
Sep. 1990 - Jan. 1993	Budapest University of Technology Faculty of Mechanical Engineering
Sep. 1988 - Sep. 1990	ELTE Apáczai Csere János High School, Budapest
Sep. 1982 - Sep. 1988	14. W. Bredel Secondary School, Berlin/GDR
Sep. 1978 - Sep. 1982	Szemere Primary School, Budapest

9.2 Related Publications

Petta, P., Staller, A., Trappl, R., Mantler, S., Szalavári, Zs., Psik, T., Gervautz, M.: Towards Engaging Full-Body Interaction, *Proceedings of 8th International Conference on Human-Computer Interaction (HCI International'99)*, August 22-27, Munich, Germany, 1999.

Schmalstieg, D., Encarnação, M., Szalavári, Zs.: Using Transparent Props For Interaction With The Virtual Table, *Proceedings of 1999 ACM Symposium on Interactive 3D Graphics (I3DG'99)*, Atlanta, GA, USA, April 26-28, 1999, pp. 147-154.

Szalavári, Zs., Eckstein, E., Gervautz, M.: Collaborative Gaming in Augmented Reality, *Proceedings of 1998 ACM Symposium on Virtual Reality Software and Technology (VRST'98)*, Taipeh, Taiwan, November 2-5, 1998, pp.195-204.

Szalavári, Zs., Gervautz, M.: Interaktion mit virtuellen Informationen in realen Umgebungen - das "Personal Interaction Panel", *Arbeiten und begreifen: Neue Mensch-Maschine-Schnittstellen*, lit-Verlag, Germany, 1998, pp. 147-158.

Faisstnauer, C., Schmalstieg, D., Szalavári, Zs., Computer-Assisted Selection of 3D Interaction and Navigation Metaphors, *Proceedings of the 1998 VRST Workshop on Computer Graphics*, Taipei, Taiwan, Nov. 5-6, 1998, p. 4.

Szalavári, Zs., Schmalstieg, D., Fuhrmann, A., Gervautz, M.: "Studierstube" – An Environment for Collaboration in Augmented Reality. *Journal of the Virtual Reality Society – Virtual Reality – Systems, Development and Applications*, Vol. 3, No. 1, 1998, pp. 37-49.

Szalavári, Zs., Gervautz, M.: Augmented Reality Enabled Work in "Studierstube", *EURO-VR Mini Conference '97*, Amsterdam, The Netherlands, November 10-11, 1997.

Szalavári, Zs.: Das "Personal Interaction Panel" - die Schiefertafel der Jahrtausendwende, *Proceedings of Workshop "Vom Bildschirm zum Handrad - Computer(be)nutzung nach der Desktop-Metapher"*, October 1997, pp. 32.

Szalavári, Zs., Gervautz, M.: Using the Personal Interaction Panel for 3D Interaction, *Proceedings of Conference on the Latest Results in Information Technology*, Budapest, Hungary, May 28-29, 1997.

Szalavári, Zs., Gervautz, M.: The Personal Interaction Panel – A Two-Handed Interface for Augmented Reality, *Computer Graphics Forum (Proceedings of EUROGRAPHICS'97)* Vol. 16, No. 3, September, 1997, pp. 335-346.

Schmalstieg, D., Szalavári, Zs., Fuhrmann, A., Gervautz, M.: "Studierstube"- An Environment for Collaboration in Augmented Reality. *Proceedings of 1996 Workshop on Collaborative Virtual Environments CVE'96*, Nottingham, UK, September 19-20, 1996, p.2.