EGVE Symposium (2008)
B. Mohler and R. van Liere (Editors)

miniCAVE—A Fully immersive Display System Using Consumer Hardware

Stefan Schlechtweg

Anhalt University of Applied Science
Department of Computer Science
Köthen, Germany
eMail: stefan.schlechtweg@inf.hs-anhalt.de

Abstract

We present the design and construction of a small scale CAVE that employs active stereo using shutter glasses for viewing as well as for projection. It is controlled by a single server with four graphics boards. The main use of this device is educational, students should be given a possibility to see their programs run in an immersive setting. Further applications include VR design studies and the development of new VR interaction techniques using the Nintendo Wii controller as input device.

Categories and Subject Descriptors (according to ACM CCS): I.3.2 [Computer Graphics]: Graphics Systems I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism

1. Introduction

Even though it is not the most active and growing market, immersive information displays gain more and more importance in various application areas. The automotive industry, for example, works with stereo projected images either on large screens or in other configurations of virtual environments to evaluate designs of new cars or to study human factors issues using virtual prototypes (Can the windshield wiper switch be reached properly?). Virtual assembly lines are used to test construction procedures for automobiles, aircrafts or other large scale devices (Will the part fit through the hole left in the chassis?). The area of information visualization starts to use stereo projected images to utilize the third dimension for a better exploration of datasets and abstract information sets. So, Virtual Reality techniques are used in a much wider application field today than even a few years ago. Besides large screens and PowerWalls, CAVE Automatic Virtual Environmens are employed frequently for this purpose.

The main contributions of this paper are:

- We show the design and construction of a small scale CAVE which is built completely from available consumer hardware.
- Our system is based on a single PC with four graphics boards and we use Wii game controllers for navigation and head tracking.
- We show the use of shutter filters to be able to use low cost LCD projectors.

We start with an introduction to the general aspects of CAVEs before we describe our goals. Afterwards we introduce our concept how it was finally realized including hardware and software issues.

2. CAVE Automatic Virtual Environment and Related Work

Virtual Reality applications have become part of many application areas: games, automotive industry, medicine, design, etc. Being a human-computer interface that allows the user to perceive a computer generated environment as reality [Hen97], Virtual Reality builds on special output and input devices for interaction. Since the visual sense is most important, a Virtual Reality device usually comprises either a large scale display or ensures that the user's visual field is completely covered with the synthetic images. Two of the most common display devices for VR are the head mounted display (HMD) and the CAVE. While the first one uses small images in close distance to the eyes, the CAVE belongs to the class of systems with large displays.

A CAVE or CAVE Automatic Virtual Environment is a multi-person, room-sized, high-resolution 3D video and audio environment invented at the Electronic Visualization Lab at the University of Illinois, Chicago in 1992 [CNSD*92]. Graphics are projected in stereo on 3 to 6 walls of a room in which the users stand and interact. Due to the size of the projections, an immersive experience of the virtual world is possible. Interaction in such environments is usually performed using data gloves or wands while the position of the interacting user is tracked so that the images can be computed as to show the user's current view.

The main problem with CAVE installations is their size. There has to be enough space to hold the CAVE itself as well as the projectors. Furthermore, the projection distances have to be taken into consideration. Due to these high demands, smaller VR devices, like HMDs, are often used. They, however do not offer multi-user possibilities. Inbetween both ends of the scale, technologies like the *ImmersaDesk* [CPS*97] or virtual engineering desks like *Pi-casso* [HDS05, HSP06] make VR available for desktop applications. In this area—namely engineering applications—interaction plays an important role. The Personal Space Station [†] is one of the most interesting solution where the user looks through a see-through mirror onto a high resolution stereoscopic display. Beneath the mirror, the user interacts with the data via highly accurate optical tracking. Such systems have their advantage especially in interactive engineering applications while they lack the immersion needed for visualization applictaion in which the user mainly navigates through a threedimensional world without much manipulation.

A similar environmet like the one presented in this paper was developed by Wegman et al. [WSV*99]. Their main focus besides the size of the CAVE was, however, interaction via voice commands. Recently, Peternier et al. [PCVT07] presented the development and implementation of a low cost CAVE based on similar goals as described in this paper. In contrast to our system, they use a cluster of four PCs for image generation. Also, their system has more than twice the size than ours. The problem of supporting different viewers in a CAVE-like environment each with his or her own correct spatial perception of the virtual environment is addressed by Blach et al. [BBH*05]. They combine polarization and shutter filters to seperate viewers and stereo channels.

As far as interaction is concerned, we step back from original Virtual Reality input devices like data gloves mainly because of their price. Instead, we use a game console controller in order to investigate the possibilities connected with this. Game controllers have previously been employed for (head) tracking in various experimental settings. Nintendo's Wii remote controller has been used, for example, by J. C Lee at the Carnegie Mellon University to enrich desktop applications and computers with touch interaction and tracking[‡]. We use WiiMotes for navigation and head tracking.

3. The miniCAVE Environment

The miniCAVE environment was originally developed to give the students a possibility to see their project results in an immersive setting. Furthermore we aim for immersive information visualization and interaction studies. Therefore, game controllers are used as interaction devices. In this section we present the goals and objectives as well as the limitations which were the basis for our development. We then describe the hardware setup and its realization.

3.1. Goals and Objectives, Limitations

To build our CAVE virtual environment we had to cope with severe space limitations. The only available room was a standard office room with an approximate size of 5m × 6m. Besides the actual CAVE environment, i. e., the support frame and the projection screens we had to take into account the projection distances of the projectors to be used. The relatively small space lead to the decision to build a single user environment where the user sits within the CAVE. The space within the CAVE was then determined by the standard size of the projection screens which in our case was 101cm × 76cm.

3.2. The Corpus and Projector Stands

A wooden frame supports the CAVE and holds the projection screens. The frame looks like a regular table without the table board. The hight of the table has to be designed in such a way that a user sitting on an office chair holds his head in a hight which corresponds to the middle of the projection screens. This brings the user's head initially in the center of the tracked space. In order to make this height adjustable for different users, either the frame could be adjusted in height or the user's seat. An adjustment of the frame hight would require a recalibration of the projections so that this option has been abandoned. The projection screens are mounted to the frame in such a way that each two screens are perpendicular to each other. The top projection screen would then need a size of 101cm × 101cm to cover the complete top opening. However, the user rarely looks further than 45 degrees above so that a regular 101cm × 76cm screen is sufficient (see Figure 1) For the top screen we use a mirror to guide the projection since mounting the projector directly above the screen is impossible due to the ceiling height.

The projector stands are also wooden constructions which are designed in such a way that the projectors are approximately in the middle of the projection screens. For the top

† See http://www.ps-tech.com/product/pss/

‡ See http://www.cs.cmu.edu/~johnny/projects/wii/

Figure 1: *The CAVE frame in its final state. The tilted mirror guides the projection onto the top wall.*

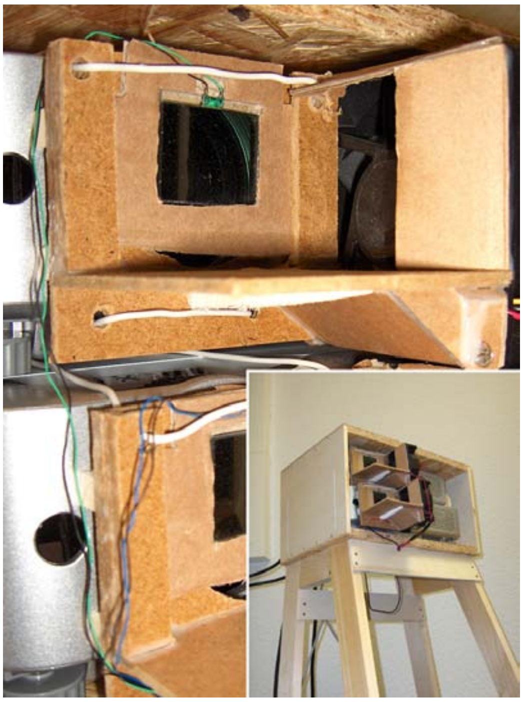


Figure 2: *The projector stand (inset) and the mounting of the shutters used for stereo viewing.*

screen the projector position has to correspond with the center of the mirror. Each projector stand holds two projectors—one for each stereo frame (cf. Section 4.1). The exact alignment of the projectors is obtained via the projectors' extendable legs as well as via the exact positionig of the projector in the stand (cf. Figure 2). The main problem with this solution is that even small vibrations to the stand and/or the projector will get the system out of calibration. However, the projectors are positioned at the sides and in the rear of the cave where normally noone will stand or walk.

3.3. Hardware

One of the main goals when designing the miniCAVE as to work with just one render server and only use available consumer class components. Our choice, therefore, was a four core processor and four high end graphics boards with two display ports each. In this way, the load will be relatively equal distributed between the processor's cores (using software which supports this) and between the graphics boards. Each graphics board will then be responsible for one stereo image pair for one wall of the CAVE.

Here is the complete list of all hardware components:

- 8 LCD projectors SANYO PLC-XU74, 1024 × 768 4:3 2500 ANSI Lumen
- 4 Screen-Tech projection screens 101cm × 76cm × 0.3cm
- shutter glasses for viewing
- 8 LCD shutters for the projectors
- 3 Nintendo Wii Remote controllers
- MSI P6N Diamond, nForce 680i SLI-Socket 775 mainboard
- 4 Geforce 8800 GTS 320MB graphics boards
- Intel Core2Quad processor
- 4 GB DDR2-800 RAM
- 250GB 16MB SATA II hard disc
- DVD writer SATA
- 2 800W ATX 2.2 power supply
- Big-Tower case

The mainboard is one of the few available boards that offers 4 PCI-Express (8x) slots which are arranged spatially in such a way that the four graphics boards can be connected. Figure 3 shows a view inside the case.

Our main concern with such a setup was the cooling and air circulation, Therefore, all available slots were equipped with fans which leads to 5 additional fans besides the processor cooling and power supply fans. The temperature values for processor and graphics boards all stay in the normal range. For power supply we have chosen two power supply units where one is responsible for the graphics boards only and the second one for the rest of the system.

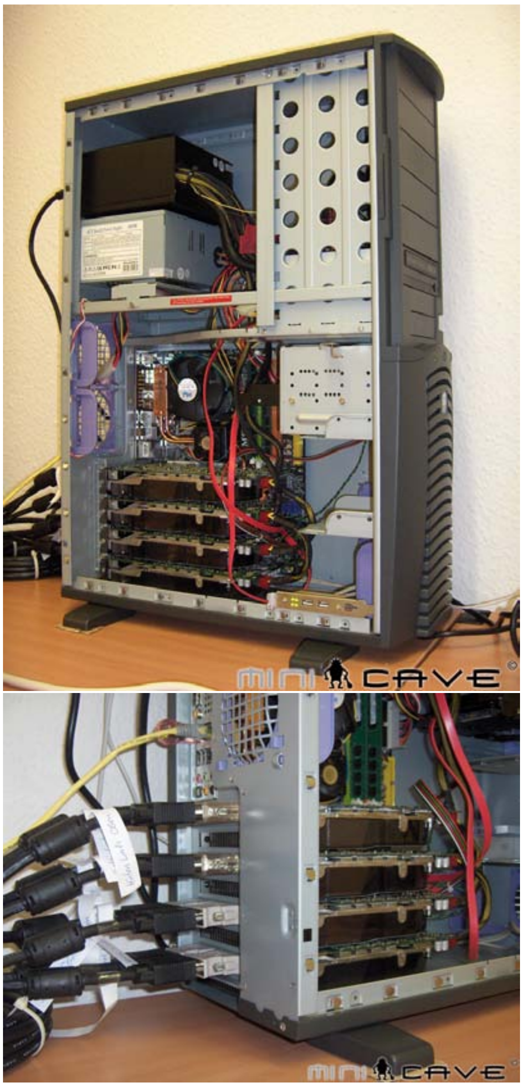


Figure 3: *The renderserver holding the four graphics boards (enlarged in bottom image).*

4. Functionality

The functionality of the miniCAVE system is mainly based on two crucial points: stereo viewing and interaction. Given the restrictions in space and budget we came up with interesting solutions for both of these points which will be discussed in the following.

4.1. Stereo Viewing

In order to achieve stereo viewing, two images from slightly different points of view (strictly speaking from the positions of each eye individually) have to be projected on the screen and it has to be ensured that the left eye only sees the "left" image and the right eye only sees the "right" image. There are mainly two possibilities for achieving a stereo view with projected images:

1. *temporal or active stereo:* The two images are projected onto the same place one after the other while ensuring that the left eye is blocked if the right image is projected and vice versa. This temporal approach typically uses shutter glasses to block the eyes alternately. The important point here is a frame rate high enough so that the immersion is still achieved. This means that images have to be projected by at least twice the rate as for monoscopic views.
2. *spatial or passive stereo:* Here, both images are projected at the same time onto the same place while the distinction of the left and right image is achieved via polarization. Different polarization filters in front of each projector lens create two differently polarized images. The user wears polarizing glasses to allow the correct image to pass to the corresponding eye. Here, the frame rate is not such a concern as with active stereo.

Both approaches have advantages and disadvantages. Passive stereo allows the use of cheaper projectors since there are no such high demands on the framerate. However, due to the polarization the user's head movement with the glasses is limited. When using linear polarization (as it is common), the user has to hold his head straight otherwise he or she would lose the stereo effect. This is a too severe limitation for an environment where the exploration of a virtual world is the main task. Active stereo, in contrast does not have this problem, but it requires high image frequencies which are not achievable with "cheap" LCD projectors.

Our idea to achieve active stereo was to keep the two projector idea which is common for passive stereo but apply shutters to the projectors instead of polarizing filters. This way we can achieve very high image frequencies since one projector doesn't have to switch between displaying the right and the left frame. The shutter "filters" we used came from rather cheap shutter glasses and were mounted in front of the projector lens. The sync signal fom a standard VGA graphics board is used to sncronize all eight projector shutters with the shutter glasses worn by the viewer. An additional cooling fan prevents the shutters to go blind due to the high temperatures caused by the projection lamp. Our setup is illustrated in Figure 2 and built for each wall. As a result we could use regular LCD projectors which do not extend our budget.

4.2. Interaction

In immersive virtual environments, interaction cannot be performed using mouse and keyboard. This would disract the user from his or her task in the virtual space. Instead, navigation and interaction have to be performed within the three-dimensional world using three-dimensional widgets which are controlled by input devices that support 6 degrees of freedom. The typical input devices, such as data glove or wands eiher are too expensive or require more space around for (optical) tracking.

In the miniCAVE, navigation and interaction is performed using a Nintendo Wii remote control ("WiiMote" for short). This device is equipped with an acceleration sensor to determine its orientation in space, a 1024×786 pixel black and white camera with an infrared filter to track infrared markers and several keys for interaction. There are also feedback channels in form of speakers and vibration but we are currently not using them. Orientation parameters can directly be read from the WiiMote's sensor data. For determining the position, two infrared light emitting diodes are placed at fixed positions in space close to the front projection screen. The camera tracking of the WiiMote is then used to compute the position of the device relative to these LEDs which is done by simple triangulation since the position of the diodes is fixed relative to the CAVE coordinate frame. This way we obtain the x, y and z position of the WiiMote. Reading the acceleration sensors yields the orientation.

This data is then used to steer a cursor in the virtual world and, thus, to navigate. Holding the WiiMote's fire button results in a forward motion through the scene which is always relative to the controller's orientation. This means that the user steers with pointing gestures through the virtual world. The navigation can be further supported by using the Wii Nunchuck in addition to the WiiMote (see Figure 4). This way the user navigates with the Nunchuck (in the left hand) in the same way as with the WiiMote. The WiiMote itself (in the right hand) is now free for other interaction tasks like picking. Due to the high flexibility of the WiiMote, other interaction types can be added easily. Also, some kind of force feedback is possible since the controller is equipped with a vibration feature that can be activated, for instance, if the cursor touches or penetrates an object. It is this flexibility that enables a wide variety of interaction styles, depending on the application at hand.

Figure 4: *Using WiiMote and Nunchuck for interaction.*

4.3. Head Tracking

For virtual environments it is of utmost importance that the visual impression is correct and consistent. This is only the case if the user gets the impression that he or she really sees with his or her own eyes what is presented in the scene. This is only possible if the viewing parameters for both eyes are computed in such a way that they match the eye's physical position relative to the CAVE's coordinate frame. For such tasks, head-tracking is typically employed.

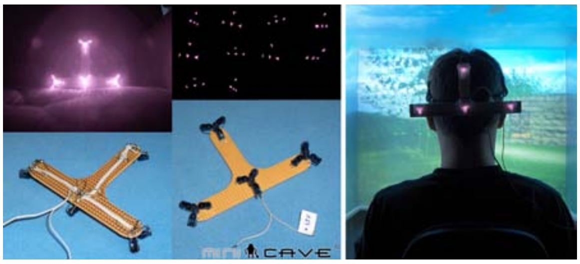
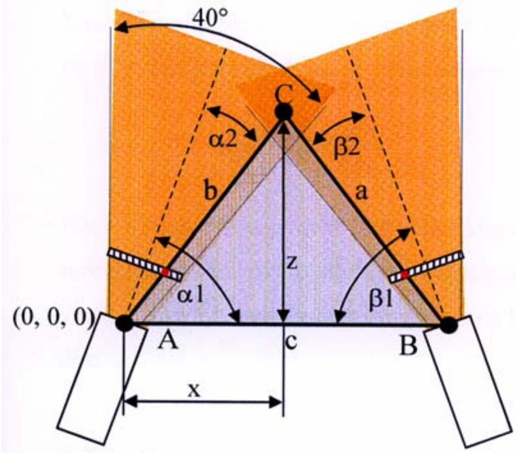


Figure 5: *Headtracking setup for the miniCAVE. Four infrared LEDs are atteched to a helmet worn by the user.*

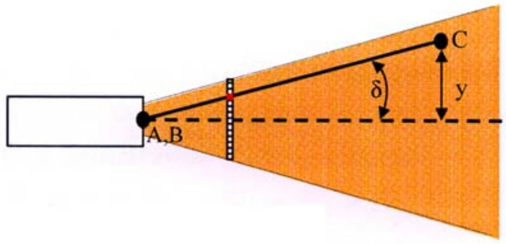
There are various ways in which the position and orientation of a user's head can be tracked. Typically the user wears some kind of sensor equipment attached to his or her head while the application then computes position and orientation based on the sensor data. While mechanical tracking was used earlier, such as with the Fakespace Boom devices, nowadays optical or electromagnetic tracking are the methods of choice for most applications. We use optical tracking due to a number of reasons: the tracking range is acceptable, the components (infrared LEDs and cameras) are within an acceptable price range and the only drawback that there has to be a clear sight between the LEDs and the camera can easily be ensured with our setup.

The user, in our case, wears a helmet which is equipped with four infrared light emitting diodes (Figure 5). The special arrangement of the diodes is later used to compute the orientation of the user's head. As sensors we, again, use two WiiMotes. The built-in cameras here are equipped with an infrared filter so that only infrared light sources are captured. Moreover, the images captured are already processed by the WiiMote so that the position of the four brightest spots in the view are returned as sensor data. These integer (x, y) pairs fall in the range between 0 and 1024 (in x direction) and between 0 and 786 (in y direction). Both cameras are positioned behind the user at fixed points. The user wears the helmet in such a way that the LED cross is at the backside of the head which is technically easier but requires some more math in the computation of the eye positions.

To compute the eye position, triangulation is used again. We first want to obtain the x and z coordinates of one marker relative to the CAVE's coordinate frame. We know the positions of the cameras in this coordinate frame as well as the positions of the marker in both images as can be seen in Figure 6(a). The triangle ΔABC between the point C to be



(a) Computation for the x and z component of the point C



(b) Computation for the y component of the point C

Figure 6: *Headtracking computation*

computed and the positions of the WiiMotes A and B is the basis for the computation. We further know the angles α_1 and β_1 as well as the distance c between the two WiiMotes. Based on the opening angles of the cameras and the positions of the marker in the images, the angles α_2 and β_2 are computed and from these the remaining parts of the triangle ΔABC, especially the point C. The y value of the marker's position (height) is easily computed using the angle between the camera's view direction and the ray through the point of the marker in the image plane (cf. Figure 6(b)). To complete the computation, an offset vector has to be added to the computed position since it describes the position of the back of the user's head. This offset compensates the distance between the markers and the respective eye position.

Now we use the special arrangement of the LEDs to compute the rotation component of the headtracking matrix as can be seen in Figure 7. From the positions of the four markers m_1 to m_4, direction vectors are determined. The directions of these vectors relative to the coordinate axes represent the rotations around the respective axes. These rotation matrices thus describe the orientation of the user's head.

The cameras of the WiiMotes have a relatively small resolution. Hence, if the marker position changes one pixel in

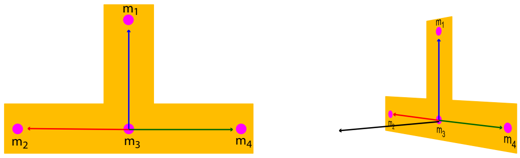


Figure 7: *The orientation vectors (red, green, blue, and black) are computed from the marker positions m_1 to m_4.*

the WiiMote's image, the marker itself might have changed its position by a rather large amount. This may yield an unsteady and jittering image. To circumvent this, we smooth the computed position data by averaging the last n values. The number of points which are averaged depends on the speed of the user's movement, i. e., the number of points in a certain radius.

The headtracking implemented this way is relatively robust even though it is not 100% exact. The smoothing of the position values and the offset vector add some error to the correct position of the user's eyes. However, we have experienced no problems when viewing a scene.

5. Software and Applications

Besides the hardware which was built to support the immersive stereo display of a virtual world and which was designed especially for a low budget and for severe space limitations, the miniCAVE environment also comprises software components.

The system itself runs on a PC with Windows Vista. Connecting the Wii remote controllers to the PC requires the use of two libraries. PPJoy[§], a joystick device driver for the Windows operating system supports virtual joysticks. Using PPJoy, each of the Wii controller's input facilities (buttons, cameras, acceleration sensors) is mapped to a virtual joystick which, in turn, can then be used in an application. A second library, GlovePIE[¶] maps the input signals of the WiiMotes to joystick input signals. The mapping can be controlled via a script language in a very flexible way.

As for now we have written a simple scene viewer for scenes which are given in VRML and other formats. This viewer is based on OpenSG since OpenSG supports clustering and multiple views and, therefore, no additional effort has to be taken to create and distribute the different views. We compute eight views, i. e., a stereo pair for each wall (left,

§ Parallel Port Joystick driver is copyrighted by Deon van der Westhuysen, see http://www.geocities.com/deonvdw/PPJoy.htm.

¶ Glove Programmable Input Emulator, copyrighted by Carl Kenner, see http://carl.kenner.googlepages.com/glovepie.

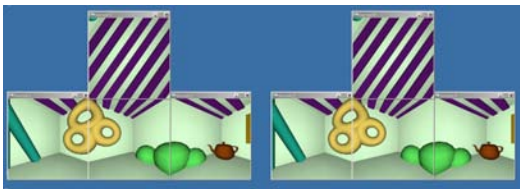


Figure 8: *Views computed*

Figure 9: *The CAVE viewer application presenting a scene in stereo mode on four walls.*

front, right, top) based on the transformations given from the headtracking and on the navigation information (see Figure 8). The viewer application simply displays the scene and the user can fly through the virtual environment using the WiiMote and Nunchuck as described earlier (see Figure 9).

Derived from the VRML viewer application, a photo browser was developed where the user can interactively browse a set of photographs. The photographs are arranged in a spiral where the order is given by some attribute (filename, date the picture was taken, etc.) Each photograph is texture mapped onto a transparent rectangle so that the images can be seen from the outside as well as from the inside. Navigation is either free or the user can fix his or her position (for example inside the spiral) and then rotate the photographs around the central axis. Picking an image brings it to the front and opens it on a larger rectangle (see Figure 10). We are planning to investigate similar applications also for exploring different kinds of information and the connections between information items.

Currently we are developing viewer applications in Java using Java3D for the rendering and another application using a game programming library. The main purpose here is educational. The miniCAVE is used at our university to support the computer graphics education and to give the students an

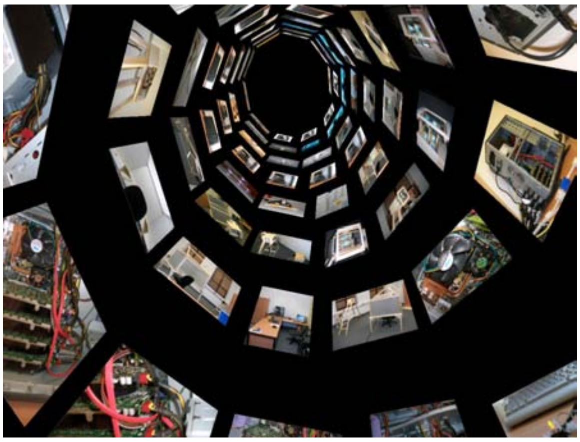


Figure 10: *The photo browser application orders the images in a spiral layout through which the user can navigate.*

opportunity to see their programs run in an immersive setting.

6. Summary

We have presented the design and construction of a low budget CAVE environment which is designed to fit in a typical office space. Even though, we have striven for a cost effective selection of hardware, today's consumer class devices offer a high quality even for immersive Virtual Reality applications. The final price for all componets was approximately 10.000 Euro (June 2007) where the projectors and the projection screens took up the biggest part. The frame and the projector stands are made in a prototypical fashion from wooden material, but can easily be constructed using standard construction elements. Care has to be taken with the projection screens. They determine the final quality of the images to a high degree and should be chosen in a way that the brightness of the projection is not reduced too much. There are also other possibilities to synchronize the projector shutters with the shutter glasses, as for instance a rotating wheel with openings, but our solution seems to be the easiest to be realized.

The most interesting parts with the setup presented in this paper are the interaction using Wii remote controllers and possible applications. Since our applications are still under development we have not yet explored the full power of the Wii controller in an immersive virtual environment. Even for such a small environment as the miniCAVE the WiiMotes offers an interesting and easy to use interface. Larger environments, as there are CAVEs and PowerWalls will even more benefit from the variuous input sensors being available. So one goal for future investigations is to investigate the usability of the Wii controllers as well as to come up with more sophisticated interaction metaphors.

The size of the miniCAVE limits or dictates the possible application areas. It is just big enough for one person to view the virtual environment and interact with it. Therefore, all applications that require full size interaction in comparison to a standing user are impossible. We therefore aim in the future for information visualization applications where the user navigates through a scene of abstract objects representing information items. Here, the sense of scale in comarison to the user's own size is not that important, we rather care for position and orientation of the elements in the scene. Other applications include design studies for cars, as an example. Here the size of the environment nicely fits the size of the study object. Other advantages of a CAVE environment, the immersion and the free movement (in a sitting position) remain with our system even under the given size limitations.

Acknowledgments

The miniCAVE would not have been possible without the help of the students who did much more for this project as it had been required. Many thanks to Marc Roßbach, Tilo Koppernock, René Kinne, Steffen Schneider and Stephan Schulz.

References

[BBH*05] BLACH R., BUES M., HOCHSTRATE J., SPRINGER J., FRÖHLICH B.: Experiences with Multi-Viewer Stereo Displays Based on LC-Shutters and Polarization. In *Proceedings of the IEEE VR Workshop Emerging Display Technologies* (2005). 2

[CNSD*92] CRUZ-NEIRA C., SANDIN D., DEFANTI T., KENYON R., HART J.: The CAVE: Audio Visual Experience Automatic Virtual Environment. *Communications of the ACM 35*, 6 (1992), 65–72. 2

[CPS*97] CZERNUSZENKO M., PAPE D., SANDIN D., DEFANTI T., DAWE T., BROWN M.: The ImmersaDesk and Infinity Wall Projection-Based Virtual Reality Displays. *Computer Graphics 31*, 2 (1997), 46–49. 2

[HDS05] HOFFMANN H., DANGELMAIER M., STEFANI O.: Mobile and Desk Integrated Virtual Reality. In *Proceedings of the International Conference on Human- Computer Interaction: HCI International 2005* (2005), vol. 7—Universal Access in HCI: Exploring New Interaction Environments. 2

[Hen97] HENNING A.: *Die andere Wirklichkeit*. Addison-Wesley-Longmann, 1997. 1

[HSP06] HOFFMANN H., STEFANI O., PATEL H.: Extending the desktop workplace by a portable virtual reality system. *International Journal of Human-Computer Studies 64*, 3 (Mar. 2006), 170–181. 2

[PCVT07] PETERNIER A., CARDIN S., VEXO F., THALMANN D.: Practical Design and Implementation of a CAVE System. In *Proceedings of the 2nd International Conference on Computer Graphics, Theory and Applications, GRAPP 2007* (2007), pp. 129–136. 2

[WSV*99] WEGMAN E. J., SYMANZIK J., VANDERSLUIS J. P., LUO Q., CAMELLI F., DZUBAY A., FU X., KHUMBAH N.-A., MOUSTAFA R. E. A., WALL R. L., ZHU Y.: The MiniCAVE: A Voice-Controlled IPT Environment. In *3rd International Immersive Projection Technology Workshop* (Heidelberg, 1999), Bullinger H.-J., Riedel O., (Eds.), Springer, pp. 179–190. 2