

Display Infrastructure for Virtual Environments (DIVE)

Thomas Wischgoll, Madison Glines, Tyler Whitlock, Bradley R. Guthrie, Corinne M. Mowrey, Pratik J. Parikh, John Flach; Wright State University; Dayton, OH

Abstract

Display systems suitable for virtual reality applications can prove useful for a variety of domains. The emergence of low-cost head-mounted displays reinvigorated the area of virtual reality significantly. However, there are still applications where full-scale CAVE-type display systems are better suited. Moreover, the cost of most CAVE-type display systems is typically rather high, thereby making it difficult to justify in a research setting.

This paper aims at providing a design of less costly display technology combined with inexpensive input devices that implements a virtual environment paradigm suitable for such full-scale visualization and simulation tasks. The focus is on cost-effective display technology that does not break a researcher's budget. The software framework utilizing these displays combines different visualization and graphics packages to create an easy-to-use software environment that can run readily on this display.

A user study was performed to evaluate the display technology and its usefulness for virtual reality tasks using an accepted measure: presence. It was found that the display technology is capable of delivering a virtual environment in which the user feels fully immersed.

Introduction

Common desktop configurations are often not very suitable for some visualization or simulation tasks. There are tasks that require higher resolution displays compared to what can be provided in a desktop setup or a higher level of immersion. Also, mouse and keyboard are not designed with 3D interaction in mind. As a result, workarounds have to be used to make these 2D input devices work within a 3D visualization or simulation environment. Using display configurations designed for virtual environments naturally provide a more intuitive input paradigm as these display setups are developed to mimic a more natural environment that the user is already familiar with. This is a benefit some visualization or simulation tasks can take advantage of as well.

Traditional virtual environments, such as original the Cave Automatic Virtual Environment (CAVE) [12], use projection-based display configurations. These display types are typically rather expensive so that they are available to only very few visualization or simulation researchers. If higher resolutions are required for the visualization or simulation task, tiled projector configurations can be used. However, these drive up the cost even further. On the other hand, similar display configurations can be achieved nowadays by using standard LCD panels. Since LCD panels are available that still provide 3D stereoscopic display capabilities, a tiled display configuration can still provide similar immersive effects than CAVE-type displays. Figure 1 shows an example of such a setup in which a user explores a molecular structure in a fully immersive environment. The advantage of this display configuration is that it is considerably less expensive

than typical CAVE-type displays. This work extends previous results [42] and includes a thorough user study for evaluating the capabilities of the display system.

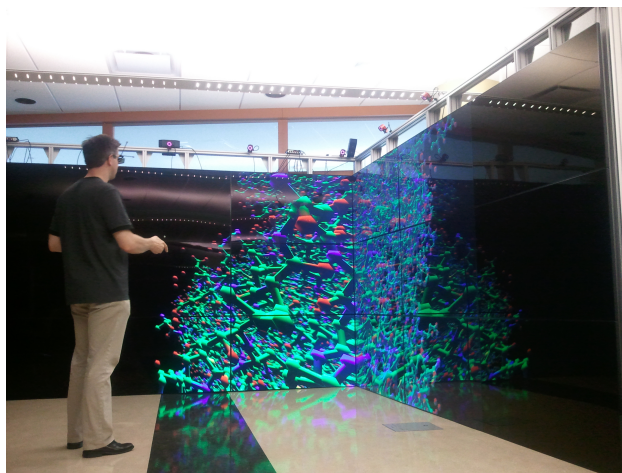


Figure 1. Display infrastructure providing a fully immersive visualization of a molecular data set.

The goal of this paper is to discuss a specific low-cost display configuration suitable for immersive virtual environments that can be utilized for different visualization and simulation tasks and is comparable with CAVE-type display systems. Basic open-source frameworks are deployed to provide a common interface to different visualization tools and libraries to make it as easy as possible for a researcher or application designer to develop their algorithms as well as running immersive simulations. The structure of this paper is as follows. First, a summary of related work is provided followed by a description of the low-cost display system suitable for immersive visualization. Then, examples for visualizations taking advantage of these displays are listed followed by an evaluation of the display system. Finally, the results are discussed and some future work is listed.

Related Work

It is well-known that head tracking, stereoscopic rendering, and a wider field of view improves the feel of presence [19]. Hence, virtual environment display systems require two major hardware components. First, display technology is necessary that allows a user to view the displayed content in 3D. Second, 3D suitable input devices are needed that do not bind the user to a certain location but instead allow for maximal freedom of movement of the user. Cheng et al. [8] provide an overview of different systems for virtual environments as well as haptic input devices. The book by Earnshaw et al. [16] provides additional details about virtual reality systems. For the display, there are typ-

ically a few different types of technologies used. Head-mounted displays (HMDs) [36, 17, 9] consist of two small screens mounted into a device that the user wears similar to a helmet such that the two screens are placed in front of the user's eyes. Since the device is equipped with two individual screens, different images for the left and right eye can be easily displayed resulting in a 3D effect experienced by the user. The resolution of head-mounted displays can be as low as only 800 by 600 pixels. Higher resolution head-mounted displays are available but can be more expensive. For example, the Open Source Virtual Reality (OSVR) HMD has a resolution of 2160×1200 for their second generation development kit. One advantage of head-mounted displays is that some can be used as see-through devices for augmented reality systems [29]. Devices currently in development following that paradigm are Microsoft's HoloLens and Magic Leap.

Other display types [10, 26, 13] rely on glasses that hide the left image from the right eye and vice versa. This allows for a majority of displays to be used. Often times large projection walls are utilized which can be configured as a large wall-type display or a CAVE-like environment. The original CAVE [12] display system provided life-like visual displays by creating a walkable display space of typically 10 by 10 feet and 90 inches tall. The walls and the floor of this display space surrounding the user consist of projection screens driven by stereo-capable projectors. For the walls typically rear-projection is used, whereas the floor often times uses front projection. Combined with active stereo glasses and a tracking system to identify the user's position, this type of configuration provides a fully immersive virtual reality system.

While CAVE systems typically use active stereo glasses, there are two different types of glasses available in combination with VR-type displays: active and passive. With passive glasses, polarization is used to ensure that the left image can only be seen by the left eye and the other way around. For projection displays, two projectors are required where a polarization filter with different polarization is placed in front of each projector. The glasses then only let light pass through that has the appropriate polarization so that each eye only sees the image generated by one of the projectors. Nowadays, even some TFT-based monitors and TV sets are becoming available that work with passive polarization glasses. Pastorelli et al. [27] describe a low-cost, projection-based display system that utilizes polarized glasses.

Active stereo glasses can actively block light from passing through, a feature that can be turned on and off for each eye separately. These types of glasses need to be synchronized with the display in such a way that ensures that the right image is only seen by the right eye and vice versa. Typically, the system displays the images for the left and right eye in an alternating fashion and activates and deactivates the glasses for the left and right eye in the active stereo glasses accordingly. The advantage of this type of glasses is that they work with many different display types, such as projection displays, CRT screens, or plasma displays.

As input devices, different wand or stylus devices are typically deployed. Often times, these are tracked either magnetically or optically to determine their position in 3D space without the need of any cabling. More recently, standard gaming devices are utilized in virtual environments which are connected wirelessly to the computer. Wischgoll et al. [43] discuss the advantages of game controllers for navigation within virtual environments. Dang et al. [14] studied the usability of various interac-

tion interfaces, such as voice, wand, pen, and sketch interfaces. Kloczek et al. [20] introduced metrics for measuring the performance when using game controllers in three-dimensional environments. Wilson et al. [41] presented a technique for entering text using a standard game controller. Glove-type devices, such as Pinch Gloves [2, 3], can also be used for intuitive input devices. Some of these devices that are currently in development allow for more detailed tracking of the fingers or force feedback.

Based on the previously described technology, a visualization of a data set can be presented to a user. For virtual simulations, some of these devices can increase the amount of immersion significantly. In order to navigate through or around a displayed model, the camera location needs to be modified. In general, a camera model describes point of view, orientation, aperture angle, and direction and ratio of motion. A general system for camera movement based on the specification of position and orientation of the camera is presented in [15], whereas Gleicher et al. [18] chose an approach where through-the-lens control by solving for the time derivatives of the camera parameters is applied. The concept of walkthroughs in simulated virtual worlds using a flying metaphor has first been explored by Brooks [4]. Other commonly applied metaphors for navigation in virtual environments (VEs) such as "eyeball in hand", "scene in hand" and "flying vehicle control" were introduced by Ware and Osborne [39].

Researchers explored the suitability of immersive display technology for visualization purposes for quite some time. Unfortunately, the high price tag of most display setups results in only a few researchers having access to high-end immersive displays. Fortunately, with the advent of VR-type devices geared toward the consumer market (mostly HMDs) prices are coming down. The possibility of using more cost-effective solutions for fully immersive display technology makes it more attractive to apply these to the area of visualization and simulation. An overview of the use of virtual reality technology for visualization can be found in the work by van Dam et al. [38] and Brooks [5]. In his book, Chaomei Chen [7] discusses the utilization of virtual environments for information visualization and provides effective examples. The availability of head-mounted displays at relatively low cost specifically is pushing this technology more and more into the consumer's hands, mostly for gaming. Comparisons between head-mounted displays and desktop-based virtual reality environments showed that user performance is similar, albeit with some tasks being performed better at the desktop [34]. Other comparative studies investigated different virtual reality induced symptoms, such as nausea, and found that head-mounted displays tend to result in an increase in those symptoms [33].

The use of tiled displays has significantly increased lately thanks to prices for display devices coming down. Tiled displays can be built using projection-based displays, standard computer monitors, or large LCD panels. Projector-based tiled displays typically require calibration to make them appear seamless and uniform across the entire image. As shown by Brown et al. [6] this calibration process can be automated. Thanks to recent advancements in graphics cards, namely ATI's Eyefinity and Nvidia's Surround technology, a single graphics card can drive up to six displays. This allows researchers to build tiled displays out of commodity off-the-shelf computers [25]. Thelen et al. [37] demonstrated a tiled display wall composed out of 50 LCD panels that are driven by 25 computers used for large-scale volume visualiza-

tion. Renambot et al. [28] introduced a scalable environment that can utilize tiled display configurations to provide a virtual high-resolution framebuffer to an application program.

There are different ways to evaluate a display system for virtual reality applications, albeit most revolve around measuring *presence* as defined as some form of realism and immersion [30]. Examples of such evaluations can be found in the works of [24] and [23]. The latter includes interaction mechanisms to expand on the analysis of the overall fidelity of the virtual reality system.

Display Infrastructure for Virtual Environments (DIVE)

This section describes an alternative display system to CAVE-type displays that maintains the advantage of a walkable immersive display system with some added benefits, specifically increased resolution and brightness. In order to devise a display system with a comparable foot-print to a CAVE-type setup, tiled display systems can be utilized. Large-screen LCD displays using LED backlighting are becoming available that have only little depth to them and some of these displays even come with very small bezels making them ideal for tiled display configurations. For example, Samsung's UA55E large-format display has only a small bezel of 3mm. To derive a tiled display configuration using these types of displays that is close to a traditional CAVE-type display, 27 of these displays can be mounted on an aluminum framing system. This results in three walls consisting of a 3x3 tiled configuration per wall. The framing system is made from 80/20 extruded aluminum with two vertical bars to support one column of displays. Figure 2 shows a technical drawing of the framing system used to mount the individual displays. There are pairs of vertical bars on which a low profile TV mount can be attached to hold the displays in place. At the front of the framing configuration there is only a single horizontal bar at the top to allow easy entrance into the display area without creating any hazard, such as stumbling.

The motivation behind developing this system was to provide a configuration that is similar to CAVE-type displays with high resolution and a maximal walkable footprint while covering the full amount of peripheral vision. This then resulted in a squared configuration to maximize the covered area. The overall walkable footprint within the display system is 12x12 feet² with a display height of 87 inches. Hence, it provides a slightly larger area at almost the same height compared to a typical CAVE configuration. Since no rear-projection is required, the overall footprint of the entire display with the framing system is only slightly larger (around 13x13 feet²).

In order to keep the computer setup driving the display close to a traditional CAVE configuration, four computers running the Linux operating system are used. Figure 3 provides an overview of the overall configuration of the computational hardware. There is one master node that provides login capabilities and shows a console-type window of the virtual environment on a standard monitor to observe what is going on within the actual display system. Three slave computers display content on the large-format displays, one dedicated to a single wall each. Obviously, these computers now need to display parts of the virtual environment on 9 individual displays. For that reason, the computers driving the walls are equipped with three ATI FirePro V7900 graphics cards combined with an ATI S400 sync card to make sure that

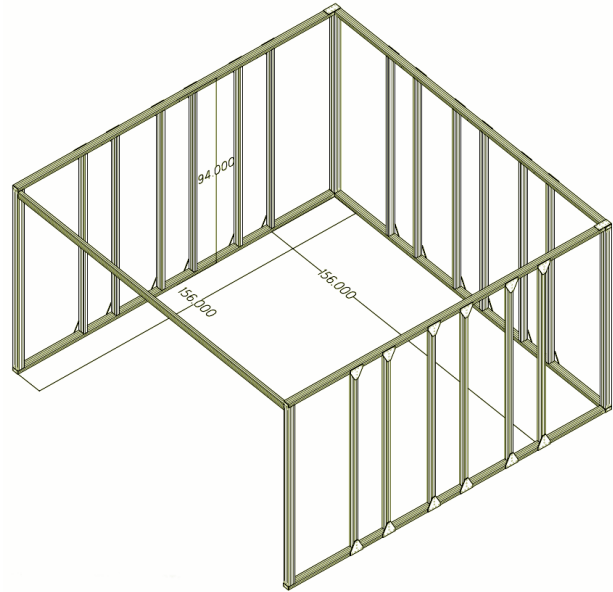


Figure 2. Framing system made out of extruded aluminum to support the individual displays.

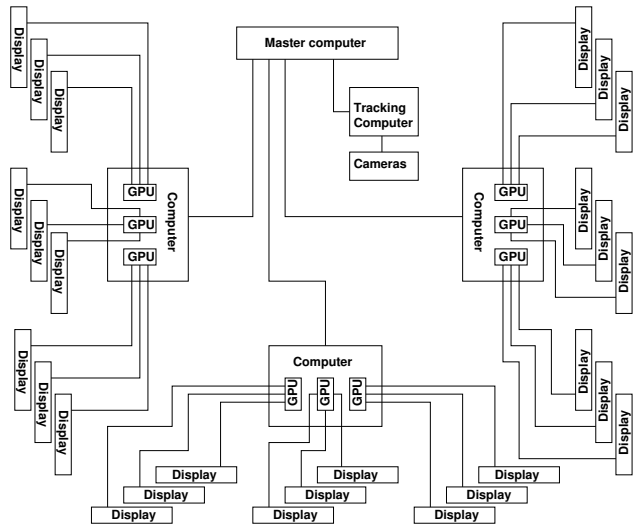


Figure 3. Schematic describing the overall setup of the DIVE system, including displays and computers involved in driving the system.

all displays show a single image at the exact same time which is required for active stereo. Three displays are then connected to a single card using ATI's Eyefinity. In the current configuration, there is a dedicated graphics card for each row of displays. This then enables the system to render to all of those three displays utilizing the left/right stereo mode provided by the HDMI 1.4 specification. Note that the full rendering performance of the graphics cards is retained as there is only one rendering step required for a row of three displays and there is a dedicated graphics card available for each of those rows of displays. The configuration of the X-server was performed via ATI's configuration tool *amdcccle*,

which also provides access to the configuration settings for the synchronization cards.

The individual displays can be controlled via the a serial port and the network. We were unable to get the serial port to communicate properly with the displays in Linux. However, the network configuration works very well. Scripts are available¹, which allow direct communication with the displays over the network. This mechanism was used to write a script that can change all the settings on the display from its default configuration, for example, the automatic 3D mode. The displays can recognize a 3D signal. Thus we devised a simple image to trick the individual displays into thinking that they receive a 3D signal and thus switch to the appropriate 3D mode. The displays can also be turned off via the network using the same script. However, this deactivates the network port so that it will not be able to be turned back on via the network. As a workaround, we simply switch to an unused input port (in our case the DVI input). Once the display recognizes the loss of input signal, it switches the display to power save mode, i.e. mostly off. Note that for this to work, the power save feature has to be enabled in the display's configuration. In order to turn the displays back on, we simply switch them all back to the HDMI input.



Figure 4. Computer system driving a single wall with three graphics card that are synced with all other computers.

To provide reliable tracking of head position and input devices, NaturalPoint's OptiTrack optical tracking system was chosen. The initial setup used the marker configuration that was supplied by NaturalPoint, i.e. a small-sized (around 2×3 inch piece of plastic with small rods to mount the reflective spheres). However, this did not yield good results. Since optical tracking requires line of sight between the cameras and the marker spheres, it is helpful if the marker spheres extend beyond the head to avoid occlusion. Figure 5 shows the configuration currently used with the display system. The marker setup is supported by wooden dowels mounted to a block of balsa wood all painted in flat black. In addition, the number of cameras was increased to eleven. This ensures that a sufficient number of marker spheres is visible at any given point in time with the entire space in between the displays

being covered.



Figure 5. Active stereo glasses with tracking spheres.

Visualization and Simulation in Virtual Environments

The visualization and simulation framework described in this paper is based on the Vrui toolkit [22] developed by Oliver Kreylos. Compared to other freely available software packages, Vrui offers more support for a variety of input devices as well as support for multi-threaded and multi-pipe rendering resulting in better rendering performance on more complex cluster-based display configurations. Vrui offers a great deal of flexibility. It can be adapted to various different types of setups ranging from fishtank VR to full-scale CAVE-type displays. In fact, the same binary can be used and based on the hostname of the computer this binary based on Vrui identifies its settings from a configuration file to match the display system. Once the configuration is set up properly, the rendering algorithm needs to be integrated into the Vrui framework. This is essentially done by adding the rendering routine to the display function of a basic Vrui sample program provided as part of the Vrui distribution following a similar paradigm than most window-managing libraries.

Vrui runs only on Linux and Mac at this point. Some tracking software, however, is only available for Windows, such as NaturalPoint's TrackingTools. To get around this issue, a dedicated tracking computer is usually used that interacts with the tracking device. NaturalPoint's TrackingTools software is capable of transmitting the tracking data over the network via the VRPN protocol.

Various types of input devices are readily supported by Vrui. The DIVE display system uses a Logitech F710 wireless gamepad as shown in figure 6 and a homebrew tracked glove. For the optical tracking, marker spheres are attached to the input devices and the input is based on positional as well as directional information. For example, one of the joysticks is tied to a forward motion. Since directional information is available via the tracking system, the forward direction is defined by the direction in which the gamepad is pointed. The data glove is also tracked by the optical tracking system using a specific marker sphere configuration that is different from the gamepad. In order to receive pinch input when the user is connecting a finger and the thumb, the internals of a wireless mouse were used where the buttons of the mouse are rerouted to metal plates on the finger tips. When connecting these

¹<https://github.com/Ape/samsungctl>



Figure 6. Gamepad with tracking spheres.

plates, this results in the mouse hardware to think that a mouse button was pushed. This then results in a signal being transmitted wirelessly to the computer that gets picked up by Vrui so that the currently running VR application can process that input in an appropriate way. Figure 7 depicts the glove currently in use for the display system.



Figure 7. Data glove with tracking spheres.

Displaying content using Vrui is rather straight forward as it follows a similar approach than most graphics packages. Any rendering code needs to be implemented in a display method that Vrui then regularly calls whenever a redraw is necessary. Since this display configurations utilizes more than one computer, one needs to be a little careful about using information that is tied directly to a specific rendering process. For example, when using a texture to map onto some geometry, this texture has to be uploaded onto each graphics card individually and their identifiers may be different. However, Vrui provides a mechanism that is capable of handling such an environment.

So far, frameworks were developed for displaying on the DIVE system that are based on plain OpenGL, OpenSceneGraph,

and VTK. Rendering using OpenGL is directly supported in Vrui as Vrui is based on OpenGL itself. Similar to other OpenGL windowing frameworks, a *display* method is provided within a Vrui application in which any OpenGL rendering code can be added. Since OpenSceneGraph and VTK are also based on OpenGL, these can be integrated into Vrui as well. However, both of these graphics packages usually rely on handling the window management and user input themselves. Obviously, Vrui already takes care of both of those two items. Hence, a workaround is required that utilizes these graphics packages but makes them render into an existing OpenGL context. In case of OpenSceneGraph, this can be done relatively easily by creating a viewer instance in which the OpenGL settings defined by Vrui are recreated followed by a traversal of the scene graph. For this, the current OpenGL modelview and projection matrices are retrieved as well as the viewport and directly written into OpenSceneGraph's viewer settings. For older versions of VTK, it is slightly more complicated as it requires the use of multipass rendering. While it does not actually require several render passes, it uses a *vtkRenderPassCollection* to force VTK to render into an existing OpenGL Drawable. Later versions of VTK (version 6.2 and higher) actually support rendering into an existing OpenGL context making this step significantly easier.

Based on these frameworks, one can benefit from most software packages that rely on OpenGL, OpenSceneGraph, or VTK. For example, the Delta3D game engine based on OpenSceneGraph is readily supported by using a variant of the OpenSceneGraph framework. Similarly, one can tie into the additional functionality provided by VTK for visualization.

Since different frameworks were developed based on VTK and OpenSceneGraph, a wide variety of applications can be supported by all the described display configurations. Obviously, virtual worlds can be created by importing realistic environments. When using OpenSceneGraph, virtual models can, for example, be imported from Google's 3D Warehouse to create realistic renditions similar to Google Earth. Such environments are frequently used by researchers from psychology to study people's behavior or use it for training purposes.



Figure 8. User exploring a volume rendering of a CT angiogram data set in an immersive environment.

Similar to rendering vascular structures, volumetric data, such as the 3D CT scans the vascular models were extracted from, can be directly displayed. An example is depicted in figure 8 showing a volume rendering of a CT angiogram of a porcine torso. The volume rendering was accomplished using the Vrui-based volume renderer Toirt Samhlaigh developed by Patrick O’Leary. By taking advantage of the navigational capabilities of the immersive environment, it becomes very easy for the user to quickly identify the appropriate perspective to identify critical sections of the volumetric data set. Similarly, the volume rendering capabilities of VTK can be used for volume rendering on the DIVE system.



Figure 9. User exploring a molecular rendering based on VRProtoshop.

Another example where immersive visualization can help investigate complex structures is the visualization of molecular structures. Figure 1 shows an example that uses VTK to derive a rendering of the molecular structure. There is a multitude of software packages available already that are suitable for visualizing molecular structures. Figure 9 shows a user navigating a molecular structure within the DIVE environment based on a Vrui version of Oliver Kreylos’ Protoshop software [21]. The DIVE configuration shows its strengths in that it is very capable of reproducing the high detail and the fine-grained structures of the molecular visualization thanks to its high resolution.

In applications such as flow visualization, immersive virtual environments can be very helpful in defining the initial condition for streamlines or streamsurfaces. Since the input devices are typically tracked, as is the case with the DIVE system, identifying 3D locations takes just a click of a button with the input controller positioned at the desired location or a touch of two fingers. Sweeping motions can similarly be used for initiating streamsurfaces. As is the case for the other examples, such immersive setups allow for more intuitive use in terms of view positions and navigating the data. We are currently in the process of porting the FAnToM software package [40] to support the Vrui environment and therefore the display system described in this paper. This will provide a vast variety of visualization algorithms applicable to flow visualizations. Particularly the DIVE display configuration can be especially useful as a result of its large display surfaces that can depict the data at higher resolution compared to many other

available systems.

There is a myriad of other visualization applications that could benefit from immersive display technology. Some Information Visualization applications may be able to benefit from the higher resolutions provided by the DIVE display setup as well as the full immersion. By reducing the overall price point, the entrance fee for researchers that want to apply this intuitive technology will be considerably reduced thereby increasing the proliferation of fully immersive display systems.

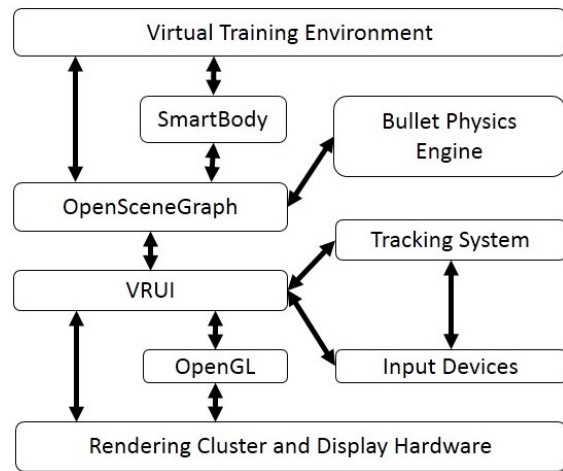


Figure 10. Architecture of the simulation environment.

Our simulation framework builds on Open Source software packages, including *SmartBody* [31] and *OpenSceneGraph* (OSG). Figure 10 outlines the architecture with all of its major components. This provides a very flexible tool that can explicitly incorporate geometric models to form a complex, computationally accessible representation of the virtual environment. Our suite of generic models make the process of generating a scenario more efficient. The software framework is designed to ingest a configuration file that describes the entire scenario in such a way that it specifies which models to incorporate into the scenario. In addition, geometric changes can be made to these models, such as scaling, rotation, and translation, so that they can be adjusted for size differences and be freely arranged to form the scenario. This facilitates an easy incorporation of available models, models that we generated ourselves, or thirdparty generated models. Additional parameters can specify whether the object to be generated is supposed to be supported as a physical object the user can move around and whether physics need to be applied. In the latter case, the physics engine *bullet* [11] is applied for a fully capable physics model, such that dropped objects fall to the ground or one can knock over other objects that then behave in a physically correct fashion.

Characters can be included in the configuration file such that a model file representing that character combined with a motion model for that character are specified. The Open Source toolkit *SmartBody* [32] then can generate a high fidelity geometric representation of that character. This toolkit also provides the ability to animate the character’s actions fully. A collision engine assures that the characters avoid objects and each other to obtain a more realistic representation overall. The characters can perform ges-

tures as well as include facial animation. Overall, this provides a great framework for generating virtual scenarios for the DIVE display systems.

Presence in the VR environment

In order to determine the quality of the experience using a specific virtual reality display technology, different metrics could be deployed. There are various issues involved from a human factors perspective when dealing with virtual environments. Stanney et al. [35], for example, provide an overview of some of these. One common way of getting a good feel for the quality of the experience with the virtual environment and the display system that is commonly accepted in the literature is to measure *presence* [44].

There are different ways of defining presence when it comes to virtual environments. Schuemie et al. [30] distinguishes between *subjective presence* and *objective presence*. The former is defined as the likelihood that a person experiences himself or herself to be physically present in the virtual environment, whereas the latter describes the likelihood of successfully completing a task.

To measure these two metrics, a virtual shopping environment was created in which study participants were asked to indicate when he or she notices specific items within the environment. The environment itself consists of sequences of shelves in different configurations mainly differing in the angles at which the shelves are oriented. In order to ensure that the focus of the study participants is solely on the display quality and issues of presence, the participants were not allowed to navigate beyond physically moving around by themselves, i.e. no motion-model using, for example, the joysticks on the gamepad, were enabled. The study participants were asked to only look around with minimal body movement. The virtual environment, however, was moved simulating some type of conveyor belt the participants are moved on through the environment. Objects 5 inches tall were placed within the shelves that were then supposed to be identified within the virtual environment. The placement of these objects was random to the participants. Based on a model describing the visibility of these objects [1], it was known which of these objects the participants should be able to see. By comparing the model data to the actual performance of the participants, a measure of objective presence can be derived. The experiment was followed directly by a questionnaire, which was done via Qualtrics in which the participants were asked about their experience to judge the participants' feel of subjective presence. A randomly sampled group of participants was recruited from within students of a human factors course resulting in 20 study participants. Additional data was collected during the experiment, such as the times when a participant noticed an object as well as the entire tracking data to reconstruct the participant's movements after the experiment.

On average, the 20 study participants were able to identify 29.44 of the objects. According to the model, they were supposed to see 31 objects. This equates to a success rate of around 95%. Due to the high resolution of the display the study participants seem to be very successful in identifying most of the objects resulting in a rather high level of objective presence.

The evaluation of the questionnaire provides some intuition on the subjective presence of the study participants. The questionnaire included common questions regarding presence and are

listed in the following:

1. In the computer generated world I had a sense of being there.
2. Somehow I felt that the virtual world surrounded me.
3. I felt like I was just perceiving pictures.
4. I did not feel present in the virtual space.
5. I had a sense of acting in the virtual space, rather than operating something from the outside.
6. How much did your experience in the virtual environment seem consistent with your real world experience.
7. How compelling was your sense of objects moving through space?
8. How completely were you able to actively survey or search the environment using vision?
9. How quickly did you adjust to the virtual environment?
10. How much did the visual display quality interfere or distract you from performing assigned tasks?

The study participants were able to answer these questions based on a Likert scale ranging from 1 through 5 where 1 meant strongly disagree and 5 represented strong agreement with the statement or question. The results were plotted as a bar chart separated into two classes. Figure 11 shows the results to questions where mostly positive answers were expected, whereas figure 12 lists responses to questions where the participants were assumed to respond with disagreement.

Discussion and Future Works

The Display Infrastructure for Virtual Environments (DIVE) was designed to resemble a typical CAVE configuration. However, the goal was to be significantly less expensive compared to traditional projector based configurations. At a cost of only \$120,000, it is considerably more affordable than traditional CAVE-type setups, which can cost upwards of \$750,000. Additional pricing information about other VR display types can be found in [42]. Since the specification of any flat panel TV does not allow for horizontal installation, no floor was realized, though. As a result, the display configuration provides a 270 degree surrounding view for full immersion. Since the bezels are very small, they are not intrusive at all and do not seem to reduce the visual quality. Compared to the Ispace, Barco's version of a CAVE-type display, the resolution is considerably higher and the large-format displays provide significantly more brightness so that the overall image quality is significantly improved over conventional projection-based systems. In order to further compare Ispace and DIVE, an additional small user study was utilized in which ten users (two female and eight male age 18 through 35) were asked to find different objects in different virtual environments running in each of those systems, i.e. Ispace and DIVE, followed by a short questionnaire. Overall, the participants were slightly faster in finding the objects in the DIVE setup. This may be attributed to the increased resolution provided by that configuration. Also, two participants noticed issues with nausea when using the systems but stated that the effect was less in the DIVE compared to the Ispace.

The data provided in the previous section suggests that the DIVE configuration provides a very good feel of presence, both in terms of subjective presence as well as objective presence. The study participants were very successful in performing the required task with a success rate of around 95%. Subjectively, they also

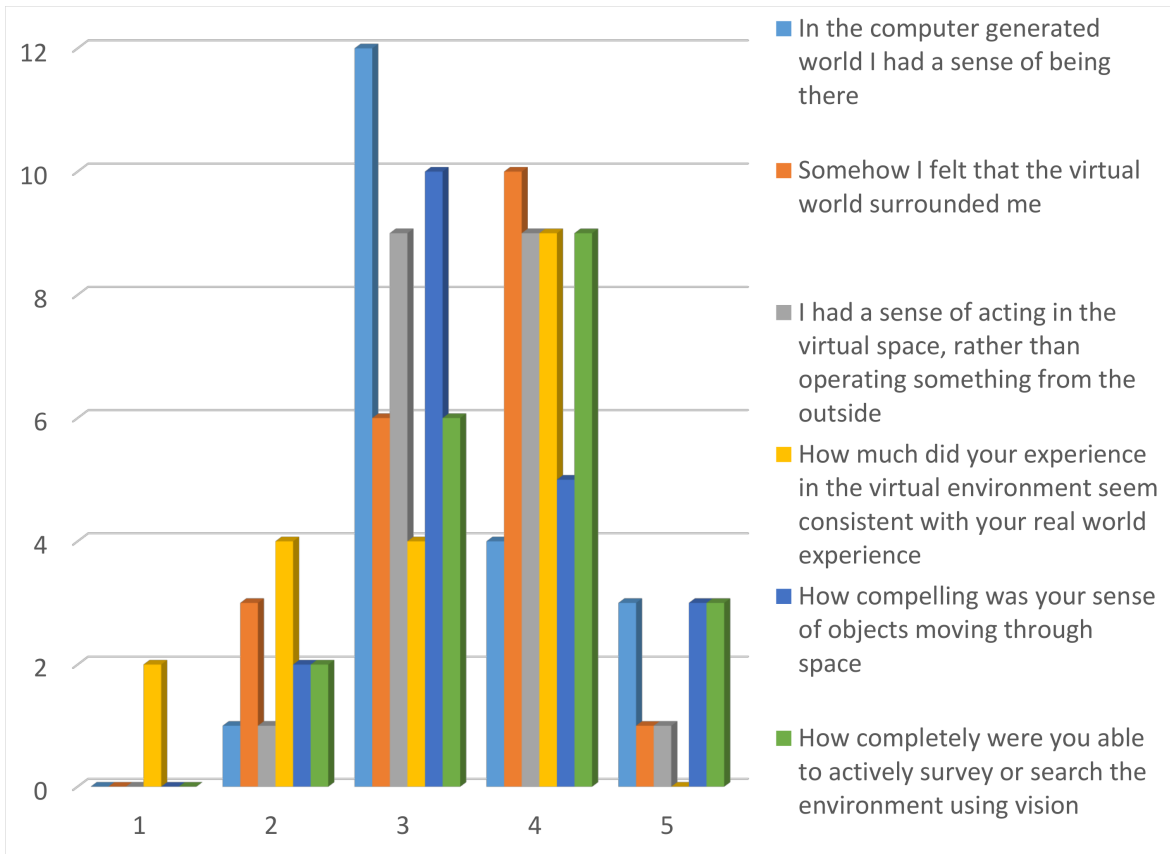


Figure 11. Responses of the study participants to questions from the questionnaire. This plot summarizes the responses with respect to questions where agreement was expected.

scored their feel of presence within the virtual environment fairly high. Questions, such as *How completely were you able to actively survey or search the environment using vision*, *How compelling was your sense of objects moving through space*, or *Somehow I felt that the virtual world surrounded me* were answered with average responses of 3.65, 3.45, and 3.45, respectively, indicating agreement with those statements. Similarly, the responses to the questions of *In the computer generated world I had a sense of being there* and *I had a sense of acting in the virtual space, rather than operating something from the outside*, which the study participated responded to on average with 3.45 and 3.50, respectively, suggest a feel of presence within the virtual environment.

Overall, the DIVE configuration performed very well, especially compared to the Ispace. The tracking system runs at a frequency of 120 Hz whereas the individual displays run at 60 Hz. The graphics cards were able to maintain that refresh rate, thereby keeping any lag at a minimum. Since lag is a contributing factor to issues with nausea this is an important aspect of the display system. Similarly, the fact that the graphics cards can still maintain a high frame rate despite the more complex hardware configuration further improves the overall visual quality as well as reduces the likelihood of issues with nausea.

In our experience, virtual environments often provide easier and more intuitive access to the visualization techniques and simulations for the untrained user. Especially, the tracked input

devices provide easier ways of selecting positions or areas in 3D, picking up an object, or navigating through 3D space. In accordance with the pros and cons of the described display systems, the utilitarian value depends on the visualization or simulation task at hand. Those tasks that require detailed and fine-grained visualizations or a high-detailed immersive environment can benefit from the increased resolution provided by the DIVE configuration. People who have used the DIVE setup so far were quite impressed with the high resolution provided by the displays as well as with the improved brightness, which is considerably higher compared to, for example, Barco's Ispace.

Acknowledgment

The author would like to thank the College of Engineering and Computer Science and the Department of Computer Science and Engineering at Wright State University for providing the funding for the display technology. This work was supported in part by NSF award 1548394.

References

- [1] Nicholas Ballester, Bradley Guthrie, Stephanie Martens, Corinne Mowrey, Pratik J Parikh, and Xinhui Zhang. Effect of retail layout on traffic density and travel distance. In *IIE Annual Conference. Proceedings*, page 798. Institute of Industrial and Systems Engineers (IISE), 2014.

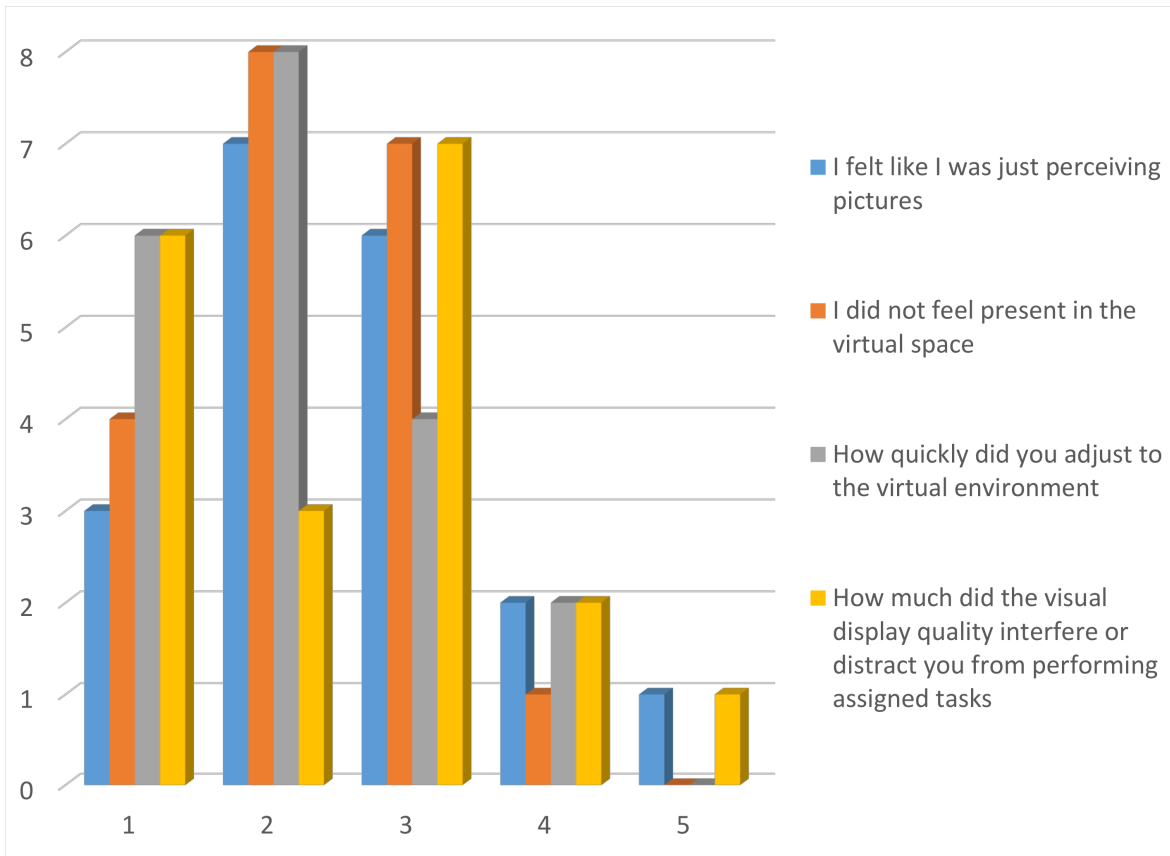


Figure 12. Responses of the study participants to questions from the questionnaire. This plot summarizes the responses with respect to questions where disagreement was expected.

- [2] D. Bowman, C. Wingrave, J. Campbell, and V. Ly. Using pinch gloves(tm) for both natural and abstract interaction techniques in virtual environments. Technical Report TR-01-23, 2001.
- [3] D. A. Bowman, C. A. Wingrave, J. M. Campbell, V. Q. Ly, and C. J. Rhoton. Novel uses of pinch gloves™ for virtual environment interaction techniques. *Virtual Reality*, 6(3):122–129, 2002.
- [4] F. Brooks Jr. Walkthrough - A dynamic graphics system for simulating virtual buildings. *Proceedings SIGGRAPH Workshop on Interactive 3D Graphics*, pages 9–21, 1986.
- [5] F.P. Brooks Jr. What’s real about virtual reality. *IEEE Computer Graphics and Applications*, 19(6):16–27, 1999.
- [6] M. Brown, A. Majumder, and R. Yang. Camera-based calibration techniques for seamless multiprojector displays. *IEEE Transactions on Visualization and Computer Graphics*, 11(2):193–206, 2005.
- [7] Chaomei Chen. *Information visualisation and virtual environments*. Springer Science & Business Media, 2013.
- [8] Yufang Cheng, Cheng-Li Huang, and Chung-Sung Yang. Using a 3d immersive virtual environment system to enhance social understanding and social skills for children with autism spectrum disorders. *Focus on Autism and Other Developmental Disabilities*, 30(4):222–236, 2015.
- [9] J. C. Chung, M. R. Harris, F. P. Brooks Jr., H. Fuchs, M. T. Kelley, J. W. Hughes, M. Ouh-Young, C. Cheung, R. L. Holloway, and M. Pique. Exploring virtual worlds with head-mounted displays. In *Proceedings SPIE Conference, Non-holographic True Three-Dimensional Display Technologies*, pages 42–52, Jan 1989.
- [10] Daniel C. Cliburn. Virtual reality for small colleges. *J. Comput. Small Coll.*, 19(4):28–38, 2004.
- [11] Erwin Coumanns. *Bullet physics library*, 2012.
- [12] Carolina Cruz-Neira, Daniel J Sandin, and Thomas A DeFanti. Surround-screen projection-based virtual reality: the design and implementation of the cave. In *Proceedings of the 20th annual conference on Computer graphics and interactive techniques*, pages 135–142. ACM, 1993.
- [13] Marek Czernuszenko, Dave Pape, Daniel Sandin, Tom DeFanti, Gregory L. Dawe, and Maxine D. Brown. The immersadesk and infinity wall projection-based virtual reality displays. *SIGGRAPH Comput. Graph.*, 31(2):46–49, 1997.
- [14] Nguyen Thong Dang, Monica Tavanti, Ivan Rankin, and Matthew Cooper. A comparison of different input devices for a 3d environment. In *ECCE '07: Proceedings of the 14th European conference on Cognitive ergonomics*, pages 153–160, New York, NY, USA, 2007. ACM.
- [15] Steven M. Drucker, Tinsley A. Galyean, and David Zeltzer. Cinema: a system for procedural camera movements. In *Proceedings of the 1992 symposium on Interactive 3D*

- graphics, pages 67–70. ACM Press, 1992.
- [16] Rae A Earnshaw. *Virtual reality systems*. Academic press, 2014.
- [17] S. Fisher, M. McGreevy, J. Humphries, and W Robinett. Virtual environment display system. In *Workshop on Interactive 3D Graphics*, pages 77–87, 1986.
- [18] Michael Gleicher and Andrew Witkin. Through-the-lens camera control. *Computer Graphics (SIGGRAPH '92 Proceedings)*, 26(2):331–340, July 1992.
- [19] Claudia Hendrix and Woodrow Barfield. Presence within virtual environments as a function of visual display parameters. *Presence: Teleoperators and Virtual Environments*, 5(3):274–289, 1996.
- [20] Chris Klochek and I. Scott MacKenzie. Performance measures of game controllers in a three-dimensional environment. In *GI '06: Proceedings of Graphics Interface 2006*, pages 73–79, Toronto, Ont., Canada, Canada, 2006. Canadian Information Processing Society.
- [21] O. Kreylos, N. Max, and S. Crivelli. Protoshop: Interactive design of protein structures. In *Proceedings of CASP5-Fifth Meeting on the Critical Assessment of Techniques for Protein Structure Prediction*, pages A213–A214, Pacific Grove, California, 2002.
- [22] Oliver Kreylos. Environment-Independent VR Development. *Lecture Notes in Computer Science*, 5358:901–912, 2008.
- [23] R. P. McMahan, D. A. Bowman, D. J. Zielinski, and R. B. Brady. Evaluating display fidelity and interaction fidelity in a virtual reality game. *IEEE Transactions on Visualization and Computer Graphics*, 18(4):626–633, April 2012.
- [24] Taeyong Moon and Gerard J. Kim. Design and evaluation of a wind display for virtual reality. In *Proceedings of the ACM Symposium on Virtual Reality Software and Technology*, VRST '04, pages 122–128, New York, NY, USA, 2004. ACM.
- [25] Nirnimesh, P. Harish, and P. J. Narayanan. Garuda: A scalable tiled display wall using commodity pcs. *IEEE Transactions on Visualization and Computer Graphics*, 13(5):864–877, 2007.
- [26] Dave Pape and Josephine Anstey. Building an affordable projective, immersive display. In *SIGGRAPH '02: ACM SIGGRAPH 2002 conference abstracts and applications*, pages 55–55, New York, NY, USA, 2002. ACM.
- [27] Emiliano Pastorelli and Heiko Herrmann. A small-scale, low-budget semi-immersive virtual environment for scientific visualization and research. *Procedia Computer Science*, 25:14–22, 2013.
- [28] L. Renambot, A. Rao, R. Singh, B. Jeong, N. Krishnaprasad, V. Vishwanath, V. Chandrasekhar, N. Schwarz, A. Spale, C. Zhang, et al. Sage: the scalable adaptive graphics environment. *Proceedings of WACE*, 9(23):2004–09, 2004.
- [29] Jannick P. Rolland and Henry Fuchs. Optical versus video see-through head-mounted displays in medical visualization. *Presence*, 9(3):287–309, 2000.
- [30] Martijn J. Schuemie, Peter van der Straaten, Merel Krijn, and Charles Van der Mast. Research on presence in vr: a survey. *Journal Cyberpsychology and Behavior*, 2(4):183–201, 2004.
- [31] Ari Shapiro. Building a character animation system. In Jan Allbeck and Petros Faloutsos, editors, *Motion in Games*, volume 7060 of *Lecture Notes in Computer Science*, pages 98–109. Springer Berlin / Heidelberg, 2011.
- [32] Ari Shapiro. Building a character animation system. *Motion in Games*, pages 98–109, 2011.
- [33] Sarah Sharples, Sue Cobb, Amanda Moody, and John R Wilson. Virtual reality induced symptoms and effects (vrise): Comparison of head mounted display (hmd), desktop and projection display systems. *Displays*, 29(2):58–69, 2008.
- [34] Beatriz Sousa Santos, Paulo Dias, Angela Pimentel, Jan-Willem Baggerman, Carlos Ferreira, Samuel Silva, and Joaquim Madeira. Head-mounted display versus desktop for 3d navigation in virtual reality: a user study. *Multimedia Tools and Applications*, 41(1):161, 2008.
- [35] Kay M. Stanney, Ronald R. Mourant, and Robert S. Kennedy. Human factors issues in virtual environments: A review of the literature. *Presence: Teleoperators and Virtual Environments*, 7(4):327–351, 1998.
- [36] I.E. Sutherland. A head-mounted three-dimensional display. In *Proc. the Fall Joint Computer Conference*, pages 757–764, 1968.
- [37] Sebastian Thelen, Joerg Meyer, Achim Ebert, and Hans Hagen. Giga-scale multiresolution volume rendering on distributed display clusters. In *HCIV'09 Proceedings of the Second IFIP WG 13.7 conference on Human-computer interaction and visualization*, pages 142–162, 2009.
- [38] Andries van Dam, Andrew S. Forsberg, David H. Laidlaw, Joseph J. LaViola, and Rosemary M. Simpson. Immersive vr for scientific visualization: A progress report. *IEEE Computer Graphics and Applications*, 20(6):26–52, 2000.
- [39] Colin Ware and Steven Osborne. Exploration and virtual camera control in virtual three dimensional environments. *Computer Graphics*, 24(2):175–183, March 1990.
- [40] Alexander Wiebel, Christoph Garth, Mario Hlawitschka, Thomas Wischgoll, and Gerik Scheuermann. FAnToM - lessons learned from design, implementation, administration, and use of a visualization system for over 10 years. In *Refactoring Visualization from Experience (ReVisE) 2009*, 2009.
- [41] Andrew D. Wilson and Maneesh Agrawala. Text entry using a dual joystick game controller. In *CHI '06: Proceedings of the SIGCHI conference on Human Factors in computing systems*, pages 475–478, New York, NY, USA, 2006. ACM.
- [42] Thomas Wischgoll. Display systems for visualization and simulation in virtual environments. In *Visualization and Data Analysis*, pages VDA391–1–VDA391–11. Electronic Imaging, 2017.
- [43] Thomas Wischgoll, Elke Moritz, and Joerg Meyer. Navigational aspects of an interactive 3d exploration system for cardiovascular structures. In *IASTED International Conference on Visualization, Imaging, and Image Processing (VIIP 2005)*, pages 721–726, 2005.
- [44] Bob G. Witmer and Michael J. Singer. Measuring presence in virtual environments: A presence questionnaire. *Presence: Teleoperators and Virtual Environments*, 7(3):225–240, 1998.