Transforming Dynallax Into
A Simultaneous 2D/3D Autostereo Workstation

Dennis Chau[a] , Tom Peterka[b] , and Jason Leigh[a] ,

[a]Electronic Visualization Laboratory, University of Illinois at Chicago
[b]Argonne National Laboratory

ABSTRACT

This work presents a novel immersive workstation environment that scientists can use for 3D data exploration and as their everyday 2D computer monitor. This implementation is based on an autostereoscopic dynamic parallax barrier 2D/3D display, interactive input devices. This paper will describe the my contributions to the hardware construction, software components, and its implementation as a platform for nanoscale materials science exploration.

Keywords: Information interfaces and presentation; Multimedia information systems; Artificial, augmented, and virtual realities.

1. INTRODUCTION

Scientific data, fueled by growth in computing power and physical sensor resolution, offer the possibility of exploring phenomena from the scale of the universe to subatomic particles. While our ability to simulate and collect scientific data is unprecedented, it brings with it new challenges for interactive exploration. Datasets are not only increasing in size, but complexity such that many attributes can be represented over complex field geometries. These geometries are frequently 3D spatial or 4D spatio-temporal, and 2D visualizations alone cannot capture the information contained in high-dimensional data.

Driven by the needs of nanoscale materials science for the exploration of structures used in energy research, modifications were made to the original Dynallax, an immersive desktop autostereoscopic workstation. One example of energy research being conducted at Argonne National Laboratory is the design of amorphous carbon structures for new anode materials in next-generation batteries. These structures, consisting of diamond cores surrounded by sheaths of graphite, provide increased area for trapping lithium ions in electrolytes. Figure 1 shows that complex models consisting of 700,000 atoms are difficult to view and understand using traditional display techniques that flatten deep structures onto a single view plane. Motivated by this example and others encountered in the United States Department of Energy's Energy Frontier Research Centers, we propose a new work environment for scientists.
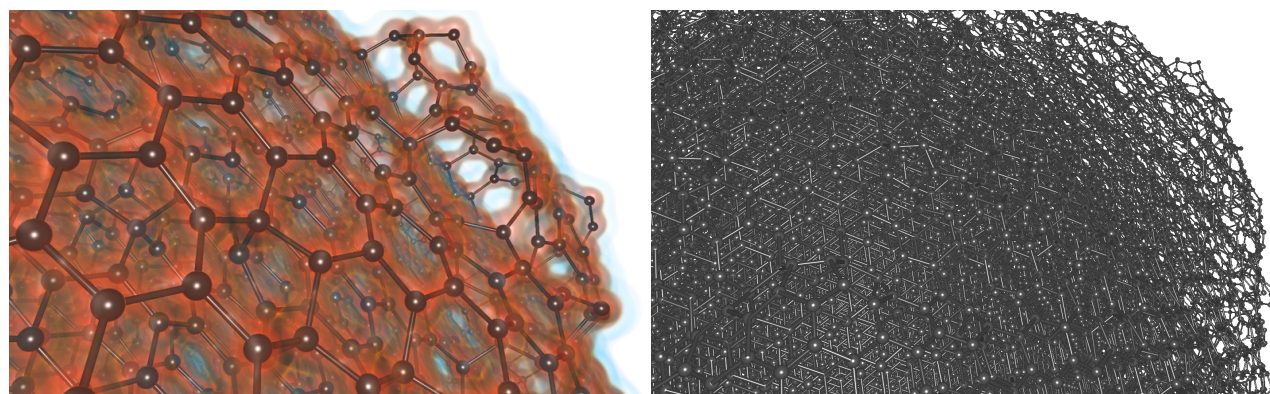

Send correspondence to dchau1@uic.edu

Figure 1. Ball-and-stick rendering without and with volume visualization. Stereo display is indispensable for visualizing such large and complex structures.

One objective when deploying the next iteration of Dynallax was to offer a transparent environment that directs the user's focus on the science problem instead of the technology. Today's science work day is fluid and consists of many interleaved tasks, so having to plan and reserve "3D time" to use high-end resources such as CAVEs can cause such facilities to be underutilized. A more beneficial methodology may be to bring the immersive environment to the scientist's desktop, where it is always on, requires minimal gear to use, and supports 3D data exploration at the same time as 2D tasks such as email and document writing. Consequently, scientists need not give up tools upon which they rely to be productive, and can maintain their existing workflows. To this end, a custom X11 window manager was integrated into an innovative autostereoscopic hardware and software.

This paper summarizes my contribution to the next generation of the Dynallax dynamic parallax barrier autostereoscopy system as a desktop form factor and a demonstration of its use in materials science. Integration of `Windy`, a custom X11 window manager into DVC, enabled simultaneous full resolution 2D content alongside 3D autostereoscopic content. This enabled material scientist from Argonne National to use this device as both a workstation and a VR platform with minimal context switching.

2. RELATED WORK

A review of earlier stereo and autostereo virtual environments shows that usability for science can be an elusive goal. Anaglyph stereo discards much of the color spectrum and potentially introduces errors for scientific visualization. Field-sequential and passive polarized stereo require glasses, which inhibit multitasking between 2D and 3D tools. Lenticular automultiscopic displays compress depth, which can also introduce errors. Varrier[1] static parallax barrier displays cannot switch between 2D and 3D mode because of the presence of a fixed parallax barrier. Though it has its own limitations, autostereoscopic displays addresses some of the issues mentioned above.

There are a number of autostereoscopic devices being developed commercially as well as in academia. One such device is the device produced by Lanman et al.[2] which has two LCDs layered, with one acting as a dynamic barrier. This is a very similar design, but Dynallax alters the barrier based on head position, where as Lanman's device alters the barrier based on the content of the scene being rendered. This solves some of the issues that are present in Dynallax system but it also introduces some new issues such as flickering. Stolle et al.[3] also introduces an autostereo device being developed in industry. Although it does not utilize a parallax barrier, it has many common goals with the Dynallax workstation including autostereo, simultaneous 2D/3D, and compatibility with existing OpenGL applications. Its main target audience is "hardcore" game players.

Lee et al.[4] draws some very interesting conclusions about the accommodation of 2D content in autostereoscopic displays. Their parallax barrier devices uses a time-multiplexed barrier to achieve a "2D only" mode. The necessity of this mode is apparent given the amount of current 2D content. Steinicke et al.[5] researchs this concept of mixed 2D/3D content on autostereo displays and refers to it as interscopic. In their research, they outline a method for mixing the 2D user interface for CAD software with a stereo rendering of the model being designed. In their research, an informal survey showed that interscopic user interface design may benefit certain technical applications. However, Dynallax device goes beyond having a separate 2D and 3D modality and offers a mixed 2D/3D paradigm, allowing users to view both types of content simultaneously.

The advances listed above and countless others have contributed to immersive stereoscopic systems and technology over the last two decades. One application area that has the potential to take advantage of these advances is domain science. For example, in their project GROPE, Brooks et al.[6] provided visualizations of molecules and their force fields to chemists trying to discern docking patterns of drugs. The authors report that because of the immersive interactions with the GROPE system, chemists had a new understanding of drug docking and receptor site interactions, increasing their situational awareness.

Another project, called Crumbs, was developed by Brady et al.[7] at the University of Illinois Urbana-Champaign for use in a CAVE[8] system. Crumbs visualized volumetric data and provided easy and intuitive ways for users to navigate and inspect the datasets. When used by biologists from various domains, they reported that Crumbs improved workflow and contributed to increased understanding of the data, acknowledging its educational possibilities.

As a final science application, Bohrer et al.[9] showed that a group of interdisciplinary scientists could collaborate in an immersive stereo environment, in this case the Duke Immersive Virtual Environment. Using a visualization, these scientists were able to explore a large complex dataset whose variables interacted in a nonlinear way. Their access to

this immersive stereo environment proved to be fundamental to synthesizing a hypothesis about the dataset. Unlike these large-scale specialty displays, Dynallax attempts to bring the advantages of these immersive stereoscopic displays to the scientist where they work, and to accommodate daily activities in addition to providing novel data exploration.

3. BACKGROUND

The work presented here is a continuation of the Dynallax system[10, 11] and the software that drives it, Dynamic View Client (DVC).[12] The original Dynallax device was a prototypes that had faded into disuse and needed to be rebuilt as seen in Figure 2. Dynallax extended the concept of a static parallax barrier. By stacking two LCD panels, one in front of the other, Dynallax can achieve a dynamic parallax barrier. DVC provides accessibility by abstracting away the complexities involved in the rendering pipeline. This section will give a brief overview of each of the topics discussed above.

3.1 Static Parallax Barrier

A parallax barrier is a periodic pattern of opaque and transparent regions placed in front of a display device. This generates parallax by making separate regions of the display visible to each eye. The regions are typically parallelograms (strips), although other shapes can also be used. In order to render the scene to the display, the left and right perspectives need to be generated. Once generated, they are spatially multiplexed into strips alternating from one left, one right, and so on. These strips are then placed in locations behind the physical barrier screen such that each eye can see only those strips which are intended for it. Parallax is achieved and the brain hopefully fuses the two parallax images into one, resulting in 3D depth perception. The barrier pattern is traditionally printed to a physical medium, such as glass in the of Varrier,[13] making it static.
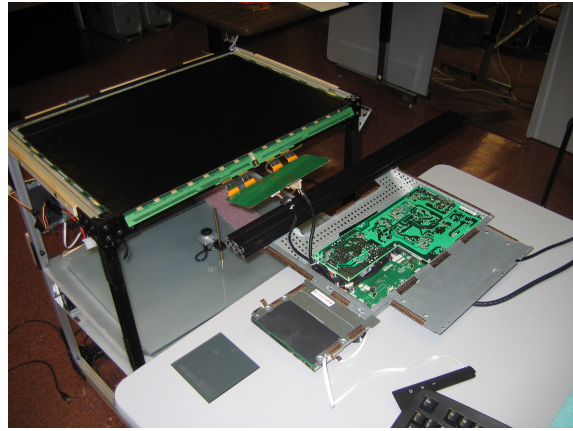

Figure 2. This is the first rebuild of the Dynallax device.

3.2 Dynamic Parallax Barrier

The Dynallax device improves on this approach by implementing a Dynamic Parallax Barrier. This is achieved by stacking two LCD displays one in front of the other. This enables the front display to generate the dynamic parallax barrier while the scene image is interleaved on the rear display, as in Figure 3. The parallax barrier are simply a pattern of black and white stripes. The semi-transparent nature of LCDs allows light to pass through on pixels rendering white. Consequently, stacking the two allows light from the rear display to pass through the white strips. The black strips are not transparent, acting exactly as the static parallax barrier did. This method is dynamic because it allows barriers of varying periods to be rendered in real time.

One key implementation note is that LCDs polarizes the light passing through it. Stacking two LCDs directly on top of each other effectively cancels out the polarization and no light passes through either LCDs. To remedy this issue, the front LCD was inverted, which allowed light to pass through. However the dual polarization of dims the display significantly even when rendering white on white. This is one avenue of future work as display and backlight technologies continue to advance.

3.3 Stereo rendering abstraction

DVC is the software library that drives the Dynallax display. It provides accessibility to users by abstracting away the complexities of the rendering pipeline. Users simply
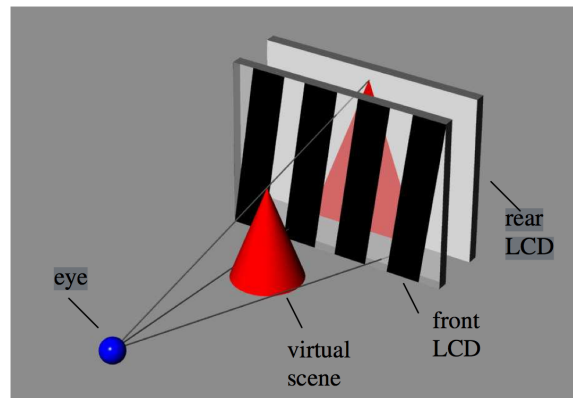

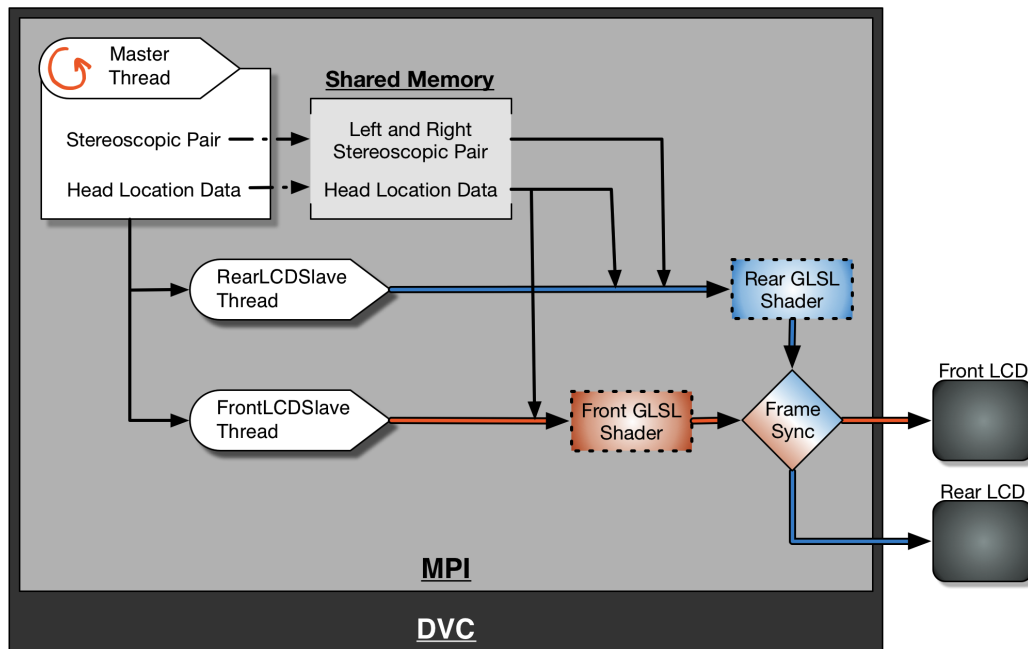Figure 3. Illustrates a parallax barrier

Figure 4. The rendering scheme of DVC is shows here. The pathway that handles the interleaving of the scene onto the rear LCD is shown in blue. The pathway that handles the linescreen rendering on the front LCD is shown in orange. These pathways represents how the autostereoscopic and parallax barrier rendering was abstracted in the original iteration of DVC.

link their applications to the DVC library and pass it stereoscopic pairs for access to the rendering pipeline. This abstraction is desired because DVC's rendering pipeline leverages OpenGL shaders and MPI[14] to maintain a dynamic parallax barrier and a corresponding interlaced stereo image. The complexities of this pipeline can be seen in Figure 4 illustrates DVC rendering pipeline.

In order to maintain accurate autostereoscopy, the front and rear display must render synchronously. DVC manages the synchronization by managing three MPI threads : Master, RearLCDSlave, and FrontLCDSlave. The Master thread initializes the MPI shared memory variables and synchronizes the other two threads within its rendering loop. The rendering loop awaits stereoscopic pairs and head location data. Both of these are stored in specified MPI shared memory locations so the other two threads have access to them. The Master thread then allows the FrontLCDSlave and RearLCDSlave to render synchronously. The FrontLCDSlave accesses the head location data only and passes it to a GLSL shader to calculate the corresponding linescreen. The resulting linescreen is texture mapped onto a quad and rendered to the front display. The RearLCDSlave accesses both the head location and the pixel buffer data. Similarly, a GLSL shader is called to interleave the scene accurately and render it onto the rear display. The calculations for both the parallax barrier and the autostereo scene interleaving were adopted from the Dynallax rendering algorithm.[10] This synchronous rendering allows both the parallax barrier and the scene to update in real-time while preserving the autostereoscpoic effect.

4. METHOD

Dynallax incorporates a hardware and software infrastructure in order to create a hybrid 2D/3D scientific workstation. The upgraded hardware infrastructure is described by the specifications and parameters for our dynamic parallax barrier display. The software infrastructure builds on the rendering pipeline to abstract away the complexities of 2D/3D windowing via a custom OS level X11 windowing manager. The final display assembly can be seen in Figure 5 and uses the Dynallax method to create 3D images by means of a dynamic parallax barrier.[10]

4.1 Hardware

To transform Dynallax into a workstation with more a desktop form factor, a stock Apple Cinema 30 inch LCD display was chosen for the rear 'interleave' screen. This display was chosen for its high (2560 x1600 pixel) resolution, brightness (400 cd/m^2) and professional image quality. The front 'barrier' screen is the LCD panel extracted from a Dell 3007WFP-HC display of the same size and resolution. The Dell's power and display components are retained in order to operate and drive the LCD panel. Its enclosure, backlight and interface electronics are not required and removed. The LCD panels are stacked face-to-face to align their interior polarization layers and allow light to pass through both panels and eliminate the need to remove any polarization layers from the bonded panel assemblies. The two LCD panels are visually aligned by pixel row using simple line patterns drawn on each LCD prior to mechanical fixing.
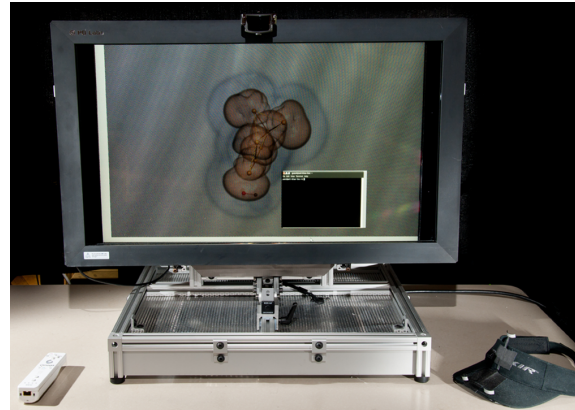

Figure 5. The Dynallax display system

The combined display assembly is mounted to an extruded aluminum framing enclosure providing for +/- 20 degree view angle adjustment. To assist in scientific discovery, lightweight user interaction devices are included. Touch capability is provided by a PQ Labs Multi-Touch 32" G3Plus touch screen, an infrared-based overlay framing the display surface. This device provides up to 32 simultaneous touches and/or gestures for application control and interaction with images. Additionally, head tracking is provided by a TrackIR v.5 tracking system, requiring users only to wear a lightweight visor and providing nonintrusive head tracking capability. A Wiimote 3D controller is also provided. User interaction is shown in Figure 6. Dynallax is powered by an Ubuntu Linux workstation with dual Intel Xeon 2.4GHz processors, two NVIDIA GTX 460 (1 GB) GPUs, and 12 GB RAM.

4.1.1 System Configuration

The front and rear screens are each connected to a different graphics card, but the screens are combined side-by-side into a single canvas using `Xinerama` mode, so that the front screen is to the right of the rear screen in the configuration.

USB drivers for the TrackIR camera and the PQLabs touch screen are not available for Linux, so we installed a virtual machine (VM) running a Windows-7 64-bit enterprise edition guest operating system (OS) on top of the Linux host OS. We selected the free VMware player over other VM software because of its support for USB devices. We set the network to bridged mode, so that the guest and host each have their own IP address, and we allotted 2GB of memory to the guest. The remaining VM settings remained at their default values.

Figure 6. User multitouch interaction

The Windows guest OS runs two utility programs that we wrote. A tracker server utility grabs raw head tracker coordinates and sends them over the loopback network interface to the Linux host on a specified port, and a touch server utility does the same for touch screen data. The virtual machine and utility programs only need to be started once and execute continuously in a minimized state. We have not experienced any noticeable lag due to using a virtual machine to drive the head tracker and touch screen.

4.1.2 Optical Parameters

The Nyquist Sampling Theorem states that each ray of light entering an eye be generated from at least two pixels, making the minimum barrier period 4 pixels wide.[13] A larger period means that more pixel samples contribute to each channel, contain larger guard regions, and further improve image quality. A larger barrier period, however, results in barrier lines that are more visible, so there is a trade off between crosstalk and barrier visibility. In our experiments, we empirically found that an ideal compromise for Dynallax is a barrier period of 6 pixels (0.060 in., or 1.5 mm). Table 1, a complete list of optical parameters, shows that we are slightly below that target in our construction, approximately 5.25 pixels.

Table 1. Major Optical Parameters	
Parameter	Value
Screen Size	25.6 in x 16.0 in (650 mm x 406 mm)
Screen Base Resolution	2560 x 1600
Effective Resolution	640 x 1600 per eye
LCD Pixel Pitch	0.010 in (0.25 mm)
Barrier Period	0.052 in (1.32 mm)
Barrier Duty Cycle	0.75 black, 0.25 white
Screen Separation	0.212 in (5.38 mm) air + 0.078 in (1.98 mm) glass
Optical Thickness	0.264 in (6.71 mm)
Optimal View Distance	24 in (609 mm)
Interocular Distance	2.5 in (63.50 mm)

From the target barrier period, the physical separation between screens is computed as follows. From Peterka et al.,[11] optical thickness t is a function of the optimal viewing distance z_{opt}, interocular distance e, and barrier period p, and $t \approx z_{opt}p/2e$. For a desktop display, we use $z_{opt} = 24$ in. (609 mm), resulting in an optical thickness of $t = 0.288$ in. (7.3 mm). To find the physical separation between the two LCD panels, the thickness of one LCD glass was subtracted, divided by its refractive index, resulting in a 0.236 in. (5.99 mm) air space between the LCDs. We then selected spacer materials that were readily available in stock thicknesses, which resulted in the parameters in Table 1.

4.2 2D/3D Window Management

The rendering pipeline of DVC needed to be modify in order to enable Dynallax to function as a 3D lens into scientific data and as a 2D workstation for everyday tasks such as web browsing as seen in Figure 7. To this end, a custom OS level window manager, `Windy`, was created. `Windy` modifies the rendering pipelines within the Master thread and the FrontLCDSlave thread in order to maintain the fidelity of the 2D content. `Windy` leverages the fact that the Dynallax device uses X11 as its primary window system. Within the Master thread, XLib,[15] a C library to interface with X11, was used to access and store all geometric information associated with 2D content. This information was later accessed by the FrontLCDSlave to bypass the barrier rendering, allowing 2D content to be displayed. The modified rendering pipeline is depicted in Figure 8. This section will go over each of these steps is greater detail.

Figure 7. Example of a 2D web browser running alongside the 3D autostereo Nanovol application.

4.2.1 Methodology of simultaneous 2D/3D

In order to render simultaneous 2D/3D content, `Windy` must operate on an OS level so that any 2D content can maintain orientation and resolution. In the main rendering loop within the Master thread, `Windy` needs to differentiate between 2D and 3D content. Once identified, the first step is to preserve orientation. Given the inverted stacking of the LCDs of the Dynallax Device, content appearing on the front LCD will appear inverted. Consequently, the front LCD must be reserved for only linesmen rendering and all 2D content must then be pushed to the rear LCD. The next step is to preserve
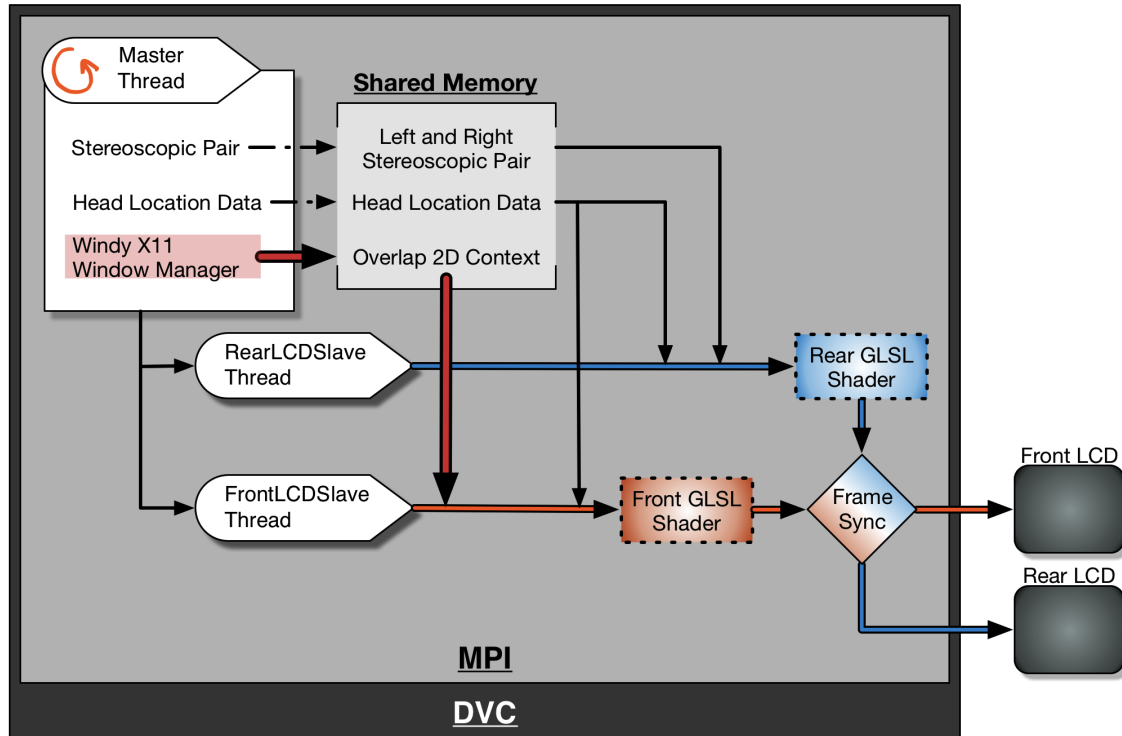
Figure 8. The modified rendering scheme of DVC is shown here. The path way in red shows how the Custom X11 Window Manger bypass and alters the parallax barrier rendering pipeline. These alterations enables 2D windows to be seen hindered by the parallax barrier.

resolution. Since 3D content interleaving is not affected by the presence of 2D content above or below 3D windows, the 3D rendering pipeline need not be changed. However 2D content that sits above a 3D window needs to have the linescreen on the front LCD cleared. Failure to do so will greatly reduce the visibility of the 2D content. Granting the FrontLCDSlave thread knowledge to the 2D content, would effective clear the linescreen on the front LCD. Given the transparent nature of a LCD when displaying white, a user can then easily see the content on the rear LCD.

4.2.2 X11 Windowing System and Xlib

In order to implement the above methodology, an understanding on the windowing hierarchy in X11 and a way of access this context is needed. At the top of the X11 hierarchy is the `Display` data structure that holds all the information associated to the device. Within `Display`, the orientation and locations of physical displays are defined as `Screens`. Since the Dynallax workstation runs in `Xinerama` mode, the two physical LCDs are defined and functions as one `Screen`. The data structure for a `Screen` defines and stores each application in an array. Each application has a `Window` entry which contains information on the applications' name, size, and location. The first element is the main `Window` and has resolution equal to the `Screen`. Their position within the array is determined by the Z depth of that application in relation to other applications.

The first step within the Master thread is to gain access to the X11 context. This can be done by gaining access to the `Display` pointer by calling `XOpenDisplay( )`. This pointer will be passed to `XScreenOfDisplay( )` in order to gain access to the `Screen` pointer. The main `Window` is extracted by passing the `Screen` pointer to `XQueryTree( )`. A `Window` pointer and an `int` pointer are passed by reference to get the root `Window` and number of children respectively. These two pieces of information will later be used to traverse the tree structure and deter the location and sizes of all the applications running.

4.2.3 Master thread : Gathering 2D Context

In order to gather the correct information about the 2D contexts, the content of the `Window` array needs to be traveled, the 2D content must be moved to the read LCD, overlapping 2D content must have its information stored and then written to MPI shared memory. A for loop is used to traverse the `Window` array given the information received from `XQueryTree()`. The properties of each `Window` can be retrieved by calling either `XGetGeometry( )` or `XGetWindowAttributes()`. All `Windows` located on the front LCD are moved by calling `XMoveWindow()`. Afterwards another pass is made through the array of `Windows`. This pass ignores all `Windows` matching geometries equal to the 3D areas defined by DVC. All others are checked to see whether it overlaps a 3D area. A 2D area overlaps if its Z-depth, is less than any 3D area. A `Window`'s Z-depth is determined by its locations within the `Windows` array. Therefore an overlapping 2D area will have a smaller index than the 3D area. 2D areas that overlap have their `Windows` properties retrieved and stored within MPI shared memory variables.

4.2.4 FrontSlaveLCD thread : Clearing the Parallax Barrier

The FrontLCDSlave thread manages the clearing of the linescreen based on the geometric information stored in MPI shared memory. The render loop of the FrontLCDSlave is primarily responsible for drawing the linescreen on the front LCD. This rendering is done via GLSL shaders by passing various arguments to the GPU that defined the properties of the linescreen. To preserve visibility, the FrontLCDSlave was modified to also pass in geometric information stored in MPI shared memory. This information corresponds to overlapping 2D content. Due to the transparent nature of LCDs, modifying the shader to drawing a white rectangle over the linescreen was sufficient. This process allows original 2D software to run unimpeded at native resolution. Additionally, since the update is done at each frame, 2D and 3D applications can be resized, moved, and switched in and out of focus seamlessly.

5. RESULTS AND CASE STUDIES

We evaluated our workstation by measuring optical parameters such as crosstalk and scattering. We then coupled Dynallax to three materials science applications that visualized isosurfaces, ball-and-stick models, and volume rendering.

5.1 Optical Tests

We evaluated the optical quality of the system using a variety of methods. The contrast ratio was measured as the difference between light emitting from both screens set to full white compared to both screens set to black, a ratio of 64:1. A test of light leakage from a white rear screen through a black front screen resulted in 2% leakage, or 98% opacity of a black front screen. We tested crosstalk using two methods; both resulted in a crosstalk ratio of 25%. The first method from Sandin et al.[13] measures the light levels in one eye channel set to black and one eye channel set to white, and computes the ratio of the black to the white intensity.
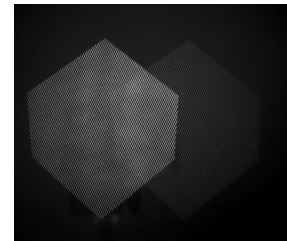

Figure 9. Photographic measurement of crosstalk. Primary channel is a brighter cube at the left and the crosstalk to its right shows a faint ghost image.

The second method relies on digital photographs of the primary and crosstalk channels for a single eye position. By photographing the same image at different shutter speeds and comparing the results, we can find the faster shutter speed that reduces the primary channel to the same intensity as the crosstalk at the original shutter speed. The ratio of these speeds is the crosstalk per eye. We conducted this experiment with the camera fitted with an 8 mm entrance pupil, to mimic the human eye, and the result is shown in Figure 9. The bright cube in the left half of this image is the primary channel, and the dim ghost on the right side is the crosstalk. The same shot taken at a four times faster exposure resulted in approximately the same intensity primary channel as the crosstalk in the original image of Figure 9.

This was a surprising finding compared to the 10% crosstalk reported in 2007.[10] We postulate that our current implementation, which is constructed from off-the-shelf components, has different optical characteristics than the implementation used previously, which was based on a product commercially designed to function as a 2-layer display. Specifically, we studied light scattering caused by what we think are internal reflections in our current design.
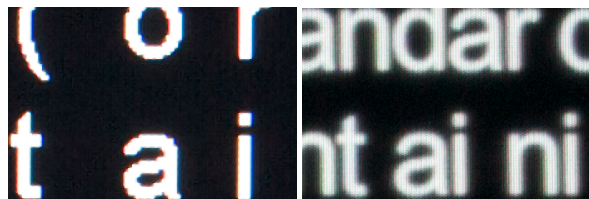
Figure 10. Comparison of text displayed in 2D on a comparable single-layer LCD monitor (left) and similar text displayed in 2D on our Dynallax system (right) reveals significant light dispersion due to internal reflections.

Figure 10 demonstrates the problem by showing plain text displayed on two monitors. A single-layer LCD display of the same model as our Dynallax rear screen is shown at left, compared to text displayed on our Dynallax display in 2D mode at right. The glow around the characters in the right image indicates significant light scattering.

To measure the percentage of scattering, we displayed a checkerboard pattern on the Dynallax display in 2D mode. Each square is 5×5 pixels, the same size as one barrier period. Light scattering appears as a gray gradient bleeding into the black squares of the photographed pattern. The original photograph is shown in the left image of Figure 11. The center image shows a Gaussian blur of radius 3 applied to the image to filter out high-frequency noise. As in the crosstalk test, we photographed a series of images at shorter exposures until the lighter squares in the darkened images matched the intensity level of the dark squares in the original image. This image is at the right of Figure 11 and corresponds to a scattering ratio of 13%, a significant contributor to the crosstalk that we observed.

5.2 Use Cases

Dynallax was employed in three applications visualizing molecular models in materials research. This field in particular benefits from stereo vision, as it frequently involves ball-and-stick models that suffer from occlusion as the number of atoms increases. To accomplish this, we coupled DVC with Nanovol, an interactive visualization tool designed for mixed ball-and-stick and volume rendering for these materials applications.[16] Since many materials volume datasets are low-resolution, Nanovol employs tricubic B-spline filtering to improve visual quality and reconstruction of sharp features. However, our autostereo framework would extend to any application that can output stereo images and receive appropriate inputs.

The first materials application involves comparison of amorphous aluminum oxide nanobowls, which serve as catalyst support structures. A molecular dynamics system simulates atomic diffusion at various temperatures. Depending on the radius of the bowl and temperature of the system, the compound may lose its structure. Domain scientists wish to examine the output of multiple simulations, and compare which parameters lead to stable or unstable bowl structures. We model this geometry as solvent-accessible molecular surfaces using SURF,[17] and use DVC to visualize the resulting triangle meshes. Conventionally, depth in these models is indicated by a RGB colormap, which can be difficult to perceive in anaglyph stereo systems. Autostereo displays allow for 3D depth perception without sacrificing color range (Figure 12).
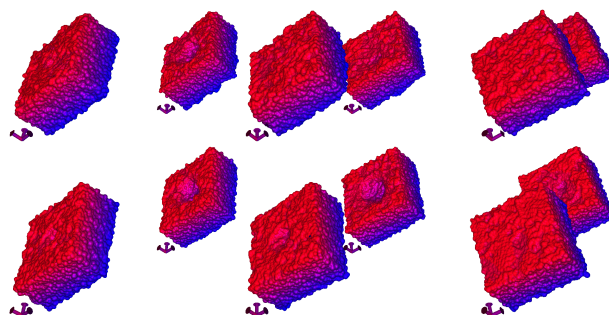

Figure 12. Comparative visualization of amorphous aluminum nanobowls demonstrates arranging and viewing models in 3D space according to different model parameters.
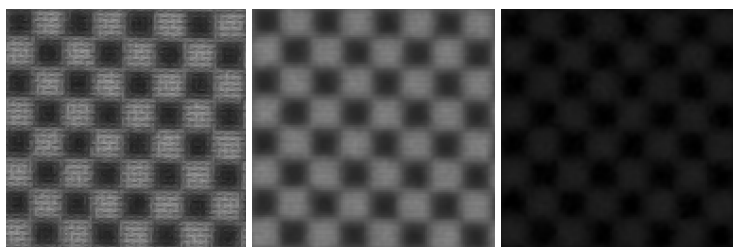

Figure 11. The amount of light scattering can be measured by displaying a checkerboard pattern in 2D mode in Dynallax and photographing at different shutter speeds. Left: original photograph shows light leaking from the white regions in a gray gradient. Center: same, with 3-pixel Gaussian low-pass filter. Right: same, taken at 13% shorter exposure.

Larger ball-and-stick models benefit strongly from stereo display. Ball-and-stick is the conventional rendering modality for most chemistry applications. However, it is often impractical to render even moderate-size models (1,000 atoms or more) in this fashion. In current materials research it is common to work with models with thousands to millions of atoms. While fog and volume rendering improve depth intuition, they obscure objects in the background. Figure 1 shows a portion of a 740,000 atom carbon nanosphere structure, with and without volume rendering, and displayed with our autostereo system.

Another use case for stereo rendering is modeling initial atom geometry prior to optimization in density functional theory (DFT) computation, for example in NWChem.[18] Domain experts spend significant time manipulating models in molecular modeling software such as Avogadro.[19] The goal is to place atoms of interest sufficiently close together that one can simulate formation of chemical bonds. Though modeling software provides some automatic optimization functionality, in practice many molecules are initially modeled by hand. Unoptimized geometry can result in nonconvergent DFT computation and wasted computation time. Due to the added depth cues, stereo display can aid in modeling initial geometry. After the DFT computation is performed, we can visualize the resulting optimized atom geometry and the electron density field, a volume dataset. An example illustration, modeling CO absorption on a platinum catalyst cluster, is shown in Figure 13.

Figure 13. CO adsorption on a platinum cluster, an example of DFT computation on manually modeled molecular geometry.

6. SUMMARY

I have presented the next interation of an autostereoscopic hardware and software solution that enables Dynallax to to transparently support scientists' daily work. Dynallax can now be used to explore data and accomplish every day office tasks without the need for a lab or dedicated space. Along with these improvements, Dynallax was also deployed in the exploration of material science models.

Our evaluation did uncover some limitations of the system. The primary limitation being: optical quality needs further improvement. Future work should address this limitation. Some suggested paths are: integrating a brighter backlight, removal of redundant polarizers on the LCD stack, using a monochrome LCD found on medical imaging displays or mitigating internal reflections in the optical path.

ACKNOWLEDGMENTS

REFERENCES

[1] Peterka, T., Kooima, R., Girado, J., Ge, J., Sandin, D., and DeFanti, T., "Evolution of the varrier autostereoscopic vr display," in [*Proceedings of IS&T / SPIE Electronic Imaging 2007*], (2007).

[2] Lanman, D., Hirsch, M., Kim, Y., and Raskar, R., "Content-adaptive parallax barriers: optimizing dual-layer 3d displays using low-rank light field factorization," in [*ACM SIGGRAPH Asia 2010 papers*], *SIGGRAPH ASIA '10*, 163:1–163:10, ACM, New York, NY, USA (2010).

[3] Stolle, H., Olaya, J.-C., Buschbeck, S., Sahm, H., and Schwerdtner, A., "Technical solutions for a full-resolution autostereoscopic 2d/3d display technology," *Stereoscopic Displays and Applications XIX* **6803**(1), 68030Q, SPIE (2008).

[4] Lee, B., Choi, H., Kim, J., Kim, Y., and Cho, S.-W., "Status and prospects of autostereoscopic 3d display technologies," in [*Lasers and Electro-Optics Society, 2007. LEOS 2007. The 20th Annual Meeting of the IEEE*], 354 –355 (oct. 2007).

[5] Stenicke, F., Ropinski, T., Bruder, G., and Hinrichs, K., "Inerscopic user interface concepts for fish tank virtual reality systems," in [*Proceedings of IEEE Virtual Reality 2007*], (2007).

[6] Brooks, Jr., F. P., Ouh-Young, M., Batter, J. J., and Jerome Kilpatrick, P., "Project gropehaptic displays for scientific visualization," in [*Proceedings of the 17th annual conference on Computer graphics and interactive techniques*], *SIGGRAPH '90*, 177–185, ACM, New York, NY, USA (1990).

[7] Brady, R., Pixton, J., Baxter, G., Moran, P., Potter, C., Carragher, B., and Belmont, A., "Crumbs: a virtual environment tracking tool for biological imaging," in [*Biomedical Visualization, 1995. Proceedings.*], 18 –25, 82 (oct-3 nov 1995).

[8] Cruz-Neira, C., Sandin, D., DeFanti, T., Kenyon, R., and Hart, J., "The cave: Audio visual experience automatic virtual environment," *Communications of the ACM* **35**(6), 64–72 (1992).

[9] Bohrer, G., Longo, M., Zielinski, D., and Brady, R., "Vr visualisation as an interdisciplinary collaborative data exploration tool for large eddy simulations of biosphere-atmosphere interactions," in [*Advances in Visual Computing*], Bebis, G., Boyle, R., Parvin, B., Koracin, D., Remagnino, P., Porikli, F., Peters, J., Klosowski, J., Arns, L., Chun, Y., Rhyne, T.-M., and Monroe, L., eds., *Lecture Notes in Computer Science* **5358**, 856–866, Springer Berlin / Heidelberg (2008).

[10] Peterka, T., Kooima, R., Girado, J., Ge, J., Sandin, D., Johnson, A., Leigh, J., Schulze, J., and DeFanti, T., "Dynallax: Solid state dynamic barrier autostereoscopic vr display," in [*Proceedings of IEEE Virtual Reality 2007*], (2007).

[11] Peterka, T., Kooima, R., Sandin, D., Johnson, A., Leigh, J., and DeFanti, T., "Advances in the dynallax solid-state dynamic parallax barrier autostereoscopic visualization display system," *IEEE Transactions on Visualization and Computer Graphics* **14**(3), 487–499 (2008).

[12] Peterka, T., Ross, R., Yu, H., Ma, K.-L., Kooima, R., and Girado, J., "Autostereoscopic display of large-scale scientific visualization," *Stereoscopic Displays and Applications XX* **7237**(1), 723706, SPIE (2009).

[13] Sandin, D., Margolis, T., Ge, J., Girado, J., Peterka, T., and DeFanti, T., "The varrier autostereoscopic virtual reality display," *ACM Transactions on Graphics, Proceedings of ACM* **24**(3), 894–903 (2005).

[14] Geist, A., Gropp, W., Huss-Lederman, S., Lumsdaine, A., Lusk, E., Saphir, W., and Skjellum, T., "Mpi-2: Extending the message-passing interface," in [*Proceedings of Euro-Par'96*], (1996).

[15] XLib: an X Window System protocol client library written in the C programming language, http://avogadro.openmolecules.net/.

[16] Knoll, A., Peterka, T., Hereld, M., Papka, M. E., Liu, B., Chan, M., and Greeley, J., "A Volumetric Framework for Registration, Analysis and Visualization of Nanostructured Materials," in [*Online Proceedings of SciDAC 2011*], SciDAC (2011).

[17] Varshney, A., Brooks, Jr., F. P., Richardson, D. C., Wright, W. V., and Manocha, D., "Defining, computing, and visualizing molecular interfaces," in [*IEEE Visualization '95 Proceedings*], Nielson, G. M. and Silver, D., eds., 36–43 (1995).

[18] Valiev, M., Bylaska, E., Govind, N., Kowalski, K., Straatsma, T., Van Dam, H., Wang, D., Nieplocha, J., Apra, E., Windus, T., et al., "Nwchem: a comprehensive and scalable open-source solution for large scale molecular simulations," *Computer Physics Communications* **181**(9), 1477–1489 (2010).

[19] Avogadro: an open-source molecular builder and visualization tool, http://avogadro.openmolecules.net/.