Dynallax: Dynamic Parallax Barrier Autostereoscopic Display

BY

THOMAS PETERKA
B.S., University of Illinois at Chicago, 1987
M.S., University of Illinois at Chicago, 2003

PH.D. DISSERTATION

Submitted as partial fulfillment of the requirements
for the degree of Doctor of Philosophy in Computer Science
in the Graduate College of the
University of Illinois at Chicago, 2007

Chicago, Illinois

This thesis is dedicated to Melinda, Chris, and Amanda, who always believed in me, and whose confidence helped me believe in myself. Graduate school would never have been possible without their love and support, and for this I will always be grateful.

ACKNOWLEDGEMENTS

I would like to express my sincere gratitude to the following individuals for their assistance and support.

- Andrew Johnson of the Electronic Visualization Laboratory (EVL), University of Illinois at Chicago (UIC) for advising me during my Ph.D. studies, for serving on the dissertation committee and for providing valuable feedback and critical reviews of my writing
- Dan Sandin of EVL, UIC for introducing me to autostereoscopy, for providing the opportunity to work with him in this exciting field, for serving on the dissertation committee, and for many years of dedication to the advancement of VR as both an art and a science
- Jason Leigh, EVL, UIC for serving on the dissertation committee and providing guidance, support, insight, and funding for my work
- Tom DeFanti, EVL, UIC, UCSD, for making time to serve on my committee, and for his ongoing support and friendship for many years at EVL
- Jurgen Schulze, UCSD, for taking the time and effort to travel from San Diego to Chicago to serve on the committee and for his interest and willingness to participate

TP

TABLE OF CONTENTS

LIST OF TABLES

TABLE PAGE

LIST OF FIGURES

FIGURE PAGE

LIST OF ABBREVIATIONS

2D, 3D	two-dimensional, three-dimensional
API	application programming interface
AS	autosteresoscopy, autostereoscopic, autostereo
CAVE	CAVE Automatic Virtual Environment
COP	center of projection
CPU	central processing unit
CRT	cathode ray tube
DLP	digital light processor
EVL	Electronic Visualization Laboratory
FLTK	FastLight Toolkit
FOV	field of view
GPU	graphics processing unit
GLSL	OpenGL shading language
GLUT	OpenGL Utility Toolkit
LC	liquid crystal
LCD	liquid crystal display
MB	megabytes
MRU	minimum resolvable unit
NYU	New York University
RAM	random access memory
RGB	red, green, blue
SDL	Simple DirectMedia Layer
UIC	University of Illinois at Chicago
UCSD	University of California at San Diego
VR	virtual reality

SUMMARY

Parallax barrier strip autostereoscopic (AS) virtual reality (VR) displays have been constructed by placing a barrier of lines in front of an LCD display device. Together with a head tracking system and associated software to interleave portions of left and right eye images in appropriate positions on the display device, an AS effect is achieved that can be quite compelling. In the past, the barrier was a static component that imposes certain limitations on the system. The barrier is manufactured to produce various system parameters such as number of viewers, type of autostereo system, size of working area. For example, in a static barrier head-tracked system the number of viewers is invariably one because of these static barrier limitations.

Substituting a dynamic parallax barrier for the static barrier frees the system from such constraints. Not only can barrier parameters be easily adjusted, but the barrier can be disabled entirely, converting the AS system to a 2D display. View distance range is expanded; moreover optimum optical conditions are maintained throughout the range. The inherent sensitivity to latency in a head-tracked parallax barrier system is mitigated by translating the dynamic barrier image in response to rapid head movements. Finally, by dynamically controlling the parallax barrier in real time, more than two eye channels are produced, for example to accommodate two viewers and provide each viewer with an independent pair of stereo perspectives.

A prototype system called Dynallax is researched and developed to demonstrate the qualities of a dynamic parallax barrier for AS. This is a small single-panel prototype based on a stacked dual-LCD display that is scalable by tiling. Besides investigating Dynallax on a purchased stacked display, improvements in dual stacked LCD construction are investigated by constructing a custom display from dissimilar components. Finally, results are shared with the scientific community through publication in a distinguished virtual reality conference.

1. INTRODUCTION

Chapter 1 introduces the dynamic parallax barrier and outlines the main areas of research. Specifically, Section 1.1 formally states the problem to be solved. Then Section 1.2 lists the advantages of the solution in terms of ease of modification of barrier parameters. Advantages continue in Section 1.3 by identifying operating modes that are possible with a dynamic barrier and that were not feasible or practical with a static one. A timeline for the study is constructed in Section 1.4, and the resources required to conduct the research are listed in Section 1.5

1.1 Problem statement

Spatially multiplexed parallax barrier autostereoscopic displays are commonly constructed using a static barrier or lenticular screen that is manufactured to produce various system output parameters. These parameters are fixed as a result of the manufacture of the barrier or screen and once constructed, cannot be changed without building a new barrier. In this research, a dynamic barrier display is constructed from a dual-layer stacked liquid crystal display (LCD) monitor. This AS method has several advantages. Various barrier parameters can be easily modified and optimized, without having to physically construct a new barrier each time, and the barrier can be updated and modified at real-time rates during system operation, resulting in new modes of operation that were not possible with a fixed barrier. The topic of this research is to experiment, understand and quantify the results from the use of a dynamic barrier in parallax barrier AS.

1.2 Features and modes

The barrier period, tilt angle, and duty cycle can all be varied dynamically. The barrier period refers to the pitch or line spacing between barrier strip lines. The lines are oriented at some non-vertical tilt angle to minimize

the effects of Moiré interference between the barrier lines and the pixel grid. The relative size of the opaque and transparent portions of one barrier period is designated by the duty cycle. Not only can the AS system be tuned by varying those parameters dynamically, but new relationships between barrier parameters and system output can be discovered and quantified because the dynamic barrier can be varied easily. The dynamic variability of the barrier affords the following modes:

2-viewer mode

By increasing the barrier period to approximately twice that in the original single viewer mode while maintaining the same physical transparent slit size, four different eye perspectives can be spatially multiplexed in the region visible through each transparent slit. The barrier period is monitored in real time to avoid conflicts between the viewing zones. The overall brightness of the display becomes approximately half as bright, requiring a bright display initially.

2D monoscopic mode

The front LCD screen can be cleared to white, allowing the rear screen to be seen unobstructed by the front screen. VR graphics can be rendered to the rear screen in ordinary 2D monoscopic mode.

View distance mode

By always computing a barrier period that is optimal for a given viewer's distance from the screen, the working view distance of the system is expanded over that of a fixed barrier display. Moreover, not only is the working range greater, but optimal view conditions are maintained over that entire range.

Rapid steering mode

The barrier pattern is rapidly translated in response to head movements such that view channels are steered and follow the viewer's head movements more closely than in a static barrier system. The effect decreases the system's sensitivity to lag.

1.3 Prototype system

In order to demonstrate the milestones outlined above, a Dynallax software system is employed. Explained in detail in Chapters 4 and 5, this software includes controls to vary system settings and modes. The AS visual output is demonstrated by both test patterns and a sample scientific visualization application. The visualization written by the author consists of a viewer for Wavefront™ ".obj" models. Some of the models are used with permission from previous visualizations developed at EVL. However, quantitative data such as ghosting, separation, etc. is gathered primarily using the test calibration pattern modes and reported in this thesis.

Initial development and testing was performed on a 3-node cluster, each node containing two processors and a single graphics card. Later, the software system was ported to a single machine with two dual-core processors and two graphics cards. The primary display device is a commodity dual stacked display, and later a custom display was constructed from components from two LCD monitors. Tracking input is provided by an existing Intersense 900 head tracker.

2. BACKGROUND

Chapter 2 reviews the prior art leading up to the current dynamic barrier research. Understandably, there is a large body of work that pre-dates this problem, ranging from topics in computer graphics, algorithms, virtual reality, stereoscopy, and autostereoscopy, to name a few. Selected topics were chosen based on their relevance to the problem and its solution. Section 2.1 begins by explaining the difference between mono, stereo, and multiscopic computer images. A few concepts of virtual reality (VR) and AS VR are covered in Section 2.2. Tracking is overviewed briefly in Section 2.3. To complete the initial introductory concepts, an overview of how to render multiple perspectives within a graphics context is presented in Section 2.4. At this point the focus remains on AS topics for the remainder of the chapter.

AS can be generated in a number of different ways, and Section 2.6 organizes the myriad methods according to type of multiplexing used in barrier, lenticular, and optical displays, and also briefly discusses volumetric and holographic displays. Rendering algorithms specific to AS literature is reviewed in Section 2.7. Optimizations to those algorithms such as exploiting coherence and graphics processing hardware appear in Section 2.8, and the chapter closes with a brief discussion of light fields and their relationship to two-view AS and multi-view AS.

2.1 Types of computer images

Computer generated imagery can be generated in monoscopic, stereoscopic, or multiscopic mode. Monoscopic mode is still most common, and is exemplified by many computer graphics packages such as OpenGL (Woo et al., 1999) and DirectX (Microsoft, 2005). A single parallel or perspective projection of a scene is generated and rendered on a display medium such as a computer monitor (perspective projection is assumed henceforth). The image may appear to be 3D due to 2D depth cues such as occlusion, foreshortening, etc. (Pfautz, 2000), but it is by definition only one view, inherently 2D. Stereoscopic imagery contains two stereo perspectives whose centers of projection are separated by an interocular distance, such that an observer whose eyes are coincident with the centers of projection of the two views sees stereo, or 3D. (provided the observer can fuse two 2D images into one 3D image) Multiscopic (or multi-view) imagery is an extension of stereoscopic imagery where more than two perspectives are produced. This allows more observers to view the imagery, and

allows the observer(s) to find multiple locations where the image can be viewed. Multiple 3D images result, one for each pair of corresponding stereo views.

The stereo and multi-view modes can be accomplished in one of two ways. (excluding hybrid combinations) Traditionally, individual perspectives are combined in either time or space. Time multiplexed, or field sequential stereo is exemplified by the CAVE (Cruz-Neira et al., 1992, 1993) and relies on rapidly displaying alternating perspectives and synchronously alternating which eye is permitted to view the image. Spatially multiplexed stereo is accomplished by simultaneously displaying all perspectives in various regions of the display, and permitting only certain regions to be seen from corresponding positions in space. Head mounted displays (HMDs) and parallax barrier methods are included in this category.

It should be noted that other multiplexing methods exist and are commonly used. For example, the C-wall (Sandin et al., 2005), Immersadesk 4 (Leigh et al., 2005) rely on polarization of light in order to display two images in the same space and allow each eye to selectively view the corresponding image using polarizing glasses. More recently, the Infitec (Jorke and Fritz, 2003) method multiplexes portions of the color spectrum, directing approximately half of the light wavelengths to each eye, again using corresponding filter glasses.

The term autostereoscopic (autostereo or AS) refers to stereoscopic or multiscopic modes that do not require glasses or attachments to be worn over the eyes in order to achieve the stereoscopic or multi-view effect. Most of the technologies listed above are stereoscopic, but only the parallax barrier methods are AS. Other AS methods, besides parallax barriers are briefly described in Section 2.6.

2.2 Virtual Reality (VR) and AS VR

AS VR requires most of the following elements. Some are clearly mandatory, such as the lack of glasses or headgear, while others are optional and even debatable, such as large size. The following definition of AS VR is

based upon some of the key elements of VR, but with the understanding that VR is a moving target for which no universally accepted definition exists. At a minimum, VR should provide first-person interactive immersion in a virtual environment. (Sherman and Craig, 2003) Additionally, it is desirable to provide stereoscopic display and a wide field of view (FOV) (Sandin et al., 2005). The definition becomes fuzzier as the list grows, but other desirable (albeit debatable) characteristics of VR are orthographic scale and perhaps multiple-viewer capability. Large size is beneficial, (Tan et al., 2004) although there is some debate about the relative importance of size vs. FOV. Of course in order for VR to become AS VR, there must be no glasses or other gear required to be worn by the user in order to see in stereo, or 3D.

AS VR is more demanding in terms of performance than other forms of VR, such as field sequential projection or HMDs. Higher numbers of views often need to be rendered at real time rates. System deficiencies can produce obvious artifacts, such as disruption of stereo manifested as cross-talk (called ghosting), pseudoscopic vision, and static and dynamic noise patterns such as guard bands and “salt and pepper” noise. Moreover, the human visual system is extremely sensitive to these effects which can cause fatigue, discomfort, headaches, eye strain, and even nausea. (Allison et al., 2001) While other VR technologies often suffer from similar technological deficiencies such as latency, low frame rates, and tracker errors, AS VR rendering artifacts are more noticeable because of the disruption of stereo and noise patterns discussed above. These imperfections are a contributing factor to the lack of broad acceptance of AS displays thus far.

Minimal performance and quality requirements for AS VR are 20-30 Hz minimum frame rate (60 Hz is always preferred), low ghosting levels (5-10%), minimal screen noise, low levels of color shifting, and the maximal use of available display resolution.

2.3 Tracking overview

Some AS methods require precise 3D (x,y,z) positional information about the location of the head and specifically the eyes within the viewing region. Often only the head is tracked and the eye positions are inferred

from a tracked reference point on the head. Those display systems that require eye position data are called tracked systems, while untracked systems are designed to operate without positional data about the user(s).

Untracked systems exist in two varieties. Some require the observer to remain in one fixed position for which the system is calibrated (the "sweet spot") while other multi-view systems or phscolograms (Sandin et al., 1989) present multiple views in a range of positions that the observer can traverse. Tracked systems by contrast allow the observer to move and constantly update the display per the tracked position. Examples of both tracked and untracked AS systems will be presented in Section 2.6.

Dynallax is an example of a tracked system, where the position of the head (and therefore the eyes) is known in real time for one or two observers. The tracking is peripheral to this research; it is assumed that technology exists that can track several head positions simultaneously in real time, and those head positions (and hence eye positions) are inputs to Dynallax. Still, it is worthwhile to briefly explain some of the different tracking technologies that are commonly employed.

Some systems require a sensor to be worn on the head, whose position is detected by electromagnetic, acoustic, or inertial means. The Ascension Technology Flock of Birds (Ascension Technology, 2005) is an example of an electromagnetic tracker, while the InterSense 900 (Intersense, 2005) is an acoustic-inertial tracker. Both provide 6 degrees of freedom, tracking both 3D position (x,y,z) and 3D rotation (roll, pitch, yaw). Both allow multiple sensors to be employed, so that several positions/rotations can be tracked simultaneously. Alternatively, camera-based systems use digital imaging and image-based algorithms to extract 3D head position from real-time video of users within the view region. Some camera-based methods require markers to be worn, but (Girado et al., 2003, 2004) does not. Camera-based methods that require no markers or sensors to be worn are best suited for AS systems, since the overriding goal is to free the user from having to wear any attachments whatsoever.

For Dynallax, an Intersense 900 is currently employed that can accommodate up to four sensors. Two sensors, each mounted on a headband, are used, one for each of two viewers. Although the Intersense has both a wired sensor and a wireless sensor, the wireless unit is not reliable and a second wired sensor is preferred. Any of the above tracking methods can serve as an input to Dynallax; the Intersense happens to be readily available and the choice of tracking technology does not affect the research in any way.

2.4 Rendering multiple perspectives via the graphics rendering pipeline

Raster computer graphics is rendered on modern graphics hardware with a graphics pipeline. Figure 1 shows an abstracted view of the rendering pipeline for common graphics programming languages.

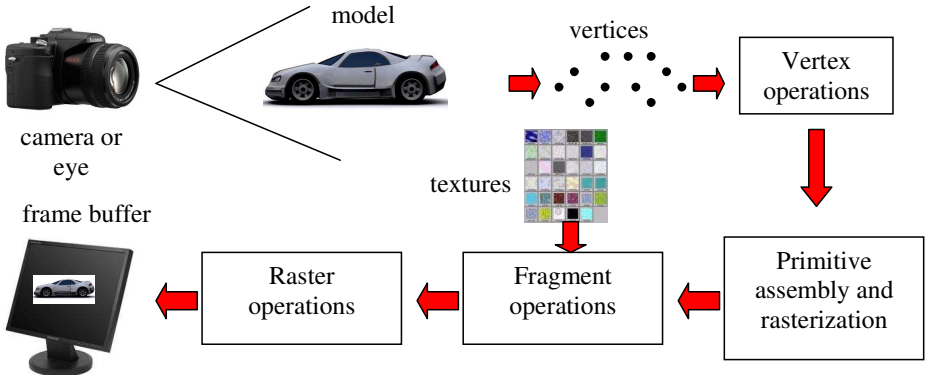


Figure 1. Raster computer graphics rendering pipeline

Whether OpenGL, DirectX or others, except for minor differences, the overall concept is the same. 3D vertex data is projected and rasterized, combined with 2D texture data, and written to the frame buffer. Recent advances in programmable shader languages such as NVIDIA CG Language (Fernando and Kilgard, 2003) or the OpenGL Shader Language (Rost, 2004) have given the programmer more control over the vertex operations and fragment operations.

For multiple views, the process must be repeated for each camera position if this model is to be retained. The model has indeed remained popular even for stereo, and it is become accepted that stereo images require twice as long to render as mono. With improvements in graphics hardware acceleration, the reduction in speed by a factor of two has become permissible, and stereo at interactive frame rates (60 Hz stereo) has become a reality for scenes of moderate complexity. (eg. order of 10^4 polygons)

AS systems commonly rely on the same approach, but AS systems are more sensitive to performance degradation. Moreover, these systems need to do more work than non-auto stereo because not only do two images need to be created, they need to be composed, multiplexed, or modulated such that the intended image reaches only the correct eye. This operation can be costly due to scene complexity and pixel-fill bandwidth. The algorithm for modulating images created by the rasterization graphics pipeline above can be written as:

a. Clear frame buffer
b. Generate full-size perspective image
c. Mask out unwanted regions
d. Modulate existing frame buffer with new image
e. Repeat from b

The time complexity is O(number of perspectives * number of pixels), ie, the frame rate is proportional to the number of views.

2.5 Introduction to parallax barrier AS

There are several popular ways to achieve AS. Each is a partial solution to the goal of AS VR, but so far no one complete solution exists. Each of the following solves certain problems while causing or ignoring others. Some of the current AS methods used are volumetric, optical, lenticular, and parallax barrier. Lenticular and parallax barrier systems can be either tracked or untracked. The parallax barrier technology is the method chosen for the Dynallax system. A brief introduction to parallax barrier and lenticular displays follows.

A parallax barrier is a periodic pattern of opaque and transparent regions placed in front of a display device that generate parallax by making separate regions of the display device visible by each eye. The regions are typically parallelograms (strips), although other shapes can also be used. The image is drawn by spatially multiplexing strips rendered from left and right eye perspectives. When those strips are 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. In Figure 2, left eye strips are drawn in blue and right eye strips in green. The parallax barrier is drawn in black and white, corresponding to opaque and transparent regions, respectively. The parallax barrier is positioned some distance in front of the display screen.

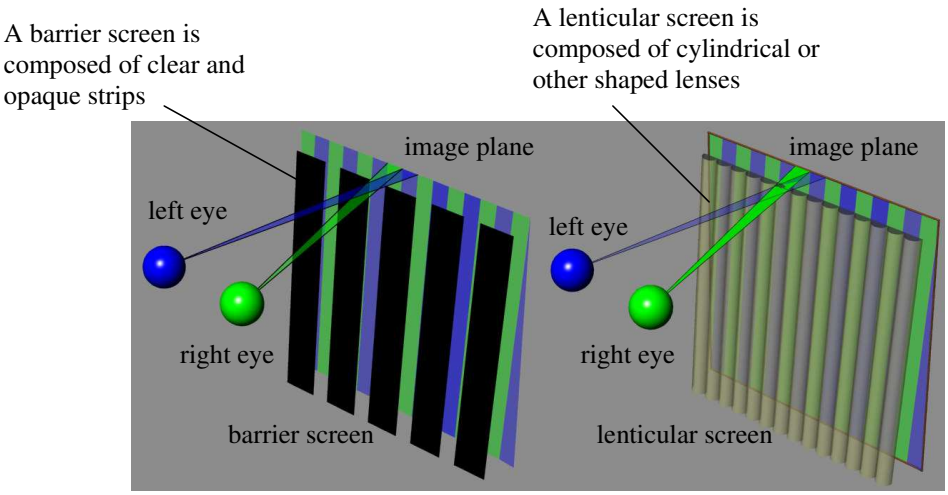


Figure 2. Parallax barrier and lenticular screen

A lenticular screen is a similar device that performs an equivalent function, but with a pattern of micro-lenses or lenticules to provide parallax. These lenses can have a variety of shapes, but most commonly cylindrical lenses are used, also shown in Figure 2. The rendering of left and right eye strips is equivalent to the parallax barrier. These methods have several advantages for AS. Among these are good focus properties for parallax barriers, good contrast, ease of construction, cost-effectiveness, compatibility with tiled displays, popularity, and

high brightness of lenticular displays. The orientation of parallax barriers and lenticular screens is often intentionally non-vertical to minimize or re-arrange Moiré patterns that result from the interference of the barrier with the display grid (Winnek, 1968), (van Berkel and Clarke, 1997), (van Berkel, 1999).

However, as in all engineering there are trade-offs, and some of the weaknesses of parallax barriers and lenticular screens are: loss of resolution, poor light efficiency (parallax barriers only), and poor focus properties of inexpensive lenticular sheets. A parallax barrier is selected for Dynallax, but it is understood that many of the advantages gained can also be realized with lenticular screens because of the nearly equivalent functions they perform. However, the dynamic requirement makes a parallax barrier a logical choice because it is possible to create and modify it electronically.

There are two popular ways to generate AS using a parallax barrier: untracked multi-view and tracked two-view. Each method has its drawbacks. Untracked multi-view AS suffers from limited resolution regardless of number of viewers present, limited performance due to many views, and noticeable transitions between views, called page flipping. Drawbacks of tracked two-view AS include support currently for only one user, and limited performance especially when higher quality rendering is performed. (quality vs. performance trade-off) Likewise, there are two methods of generating images for the above approaches. The first of these will be termed the sorting method, which is image-based and literally cuts and pastes strips of images together in interleaved order. Its major limitation is performance when the number of views becomes large. The alternative is termed the Varrier method. (Sandin et al., 2001, 2005) Unlike sorting, it is a floating point geometry-based method. Its major limitations are performance costs when high quality is required, and lack of support for multiple views or viewers.

2.6 AS systems literature review

Current literature representing the state of the art in AS systems research is surveyed in this section. This survey has 2 purposes. First, hopefully the reader will gain a sense of context surrounding the problem at hand;

dynamic parallax barrier technology is an outgrowth of much AS research that came before. Second, it is insightful to see just how many different approaches to AS exist. Space is limited to cover them all, but even a glimpse at a few should convey the sense that AS is a popular and growing field with a variety of ingenious ideas to produce autostereoscopic vision.

The first broad organization of the myriad of systems is by multiplexing method. Clearly some form of combining of images must exist in AS; these are all considered under the broad term multiplexing. Three main methods exist: time, space, and what will be termed 'other'. Space-only mutliplexed systems are covered in Secion 2.6.1, while Section 2.6.2 contrasts that with systems that use a combination of time-space multiplexing. The 'other' category in Section 2.6.3 consists of systems that do not use time or space multiplexing in the traditional sense, but rely on entirely different methods such as volumetric and holographic displays.

2.6.1 Spatial multiplexing

In space-only multiplexing, stereo or multi-view images are displayed in different regions of the display space and/or directed to different regions of the view space. Display space refers to the 2D area of the display technology such as the LCD, while view space refers to the 3D volume where the viewer is located. In space-only multiplexing, imagery is constant over time, at least with respect to the AS effect. It can vary over time due to changes in viewer position or scene content, but does not alternate rapidly between stereo perspectives over time. The Varrier system (Sandin et al., 2001, 2005) is an example of space-only multiplexing; most other barrier strip and lenticular displays fall into this category as well. A walk through the exhibit area of any major computer graphics or display technologies conference reveals many vendors marketing these devices because they are relatively simple and inexpensive to build. Usually the only required physical modification to the display device is the addition of a parallax barrier or lenticular screen to the front of an existing monitor.

Besides Varrier, other space-only systems that have published research literature are the Sharp system (Montgomery et al., 2001) (Woodgate et al., 2000), the SynthaGram (Lipton and Feldman, 2002), various integral imaging systems such as (Ren et al., 2003), the 4D-Vision system (Relke and Riemann, 2003) (Schmidt and Grasnick, 2002), and (Kleinberger et al., 2003). The Sharp display is unique in that it uses a rear dynamic barrier whose polarization can be changed electronically, resulting in a selectable 3D / 2D display. It is a single-user LCD untracked system. The SynthaGram display by StereoGraphics is popular and features a static lenticular barrier in front of an LCD display and generates a panorama of 9 views for an untracked viewer. Integral imaging systems feature not only an AS output device, but parallel input devices to capture AS images. They are also noteworthy because they often incorporate parallax in two dimensions. That is, views vary panoramically both side-to-side and up-and down. Integral imaging systems can also be space-only multiplexed, as in the system by Ren et al., which features a lenticular micro-lens sheet and strives to minimize the transition between panoramic views. The 4-D Vision system uses a wavelength-selective barrier to pass or retard different combinations of colors in order to effect parallax, and produces up to 40 views for an untracked viewer. Kleinberger's system is a single viewer, tracked system that produces a pair of stereo views using a parallax barrier in front of an LCD display.

Several conclusions can be drawn from this small sampling of spatially multiplexed systems. First, spatial multiplexing is a popular strategy, both in research and in the broader consumer market. Second, one of two design patterns is used. The system either supports a tracked single viewer or a panorama of untracked views. Up until now, the only way to accommodate multiple viewers was by using the untracked multi-view panorama.

2.6.2 Temporal – spatial multiplexing

In contrast to the space-only multiplexing approach, some AS systems rely on both time and space multiplexing. These systems not only direct a separate image to each eye position (space multiplexing), but alternate between images rapidly over time. Three different examples are presented: the Cambridge display

(Dodgson et al., 2000) (Dodgson 2002) (Cossairt et al., 2004), the New York University (NYU) display (Perlin et al. 2000, 2001), and Jung's integral imaging system. (Jung et al., 2003).

In the Cambridge display, up to 15 panoramic views per eye are drawn rapidly in succession at the full resolution of the display device. A ferro-electric shutter synchronized with the display device directs each image to a different location in view space. The result is an untracked panoramic system. In the NYU system, a course-pitch parallax barrier is constructed from a micro-stripe screen, which dynamically adjusts to the viewer's position. To make the coarse-pitch parallax barrier appear "invisible," it is varied rapidly in time in synchronization with the display device. The result is a single viewer tracked two-view system. The integral imaging system by Jung et al. incorporates a dynamically switchable array of micro-lenses that provides parallax in 2 directions and can produce multiple untracked views.

To summarize the time-space multiplexing systems, time multiplexing serves to recover resolution that otherwise must be divided among views in space-only multiplexing systems. The parallax barrier, lenticular screen, or other optical steering mechanism must be dynamically switchable. Once again however, the same design pattern emerges: either multiple panoramic views are untracked or a single user with two views is tracked. High speed display devices such as fast CRTs or DLP projection engines are required.

2.6.3 Volumetric and holographic displays

The "other" category includes primarily volumetric and holographic displays. These are not the focus of this thesis but are mentioned here for completeness because they form a significant group and deserve mention. This category also includes some of the more unconventional ideas. The grouping is somewhat arbitrary, as it is used as the "catch-all" for those ideas that do not fit neatly into the first two groups. It includes volumetric displays such as the DepthCube (Sullivan, 2004), swept volume displays such as (Miyazaki et al., 2003) and the Perspecta

display (Favalora and Lewis, 2003), static volume displays such as Solid Felix (Langhans et al., 2003) and holographic displays (Lucente and Galyean, 1995).

The DepthCube by LightSpace Technologies features a stack of 20 electrically switchable liquid crystal (LC) shutters onto which images are rapidly projected from a digital light processor (DLP) projection engine. A 3D image is projected by rapidly switching between projecting 20 slices of the 3D scene and synchronously capturing each slice on one of the 20 LC planes. The result is an untracked multiple viewer system where images are displayed in "true" depth provided they fall within the physical size of the unit. The swept volume displays feature either a rotating mirror or rotating projection screen. By synchronously projecting images onto the rotating medium, a locus of 3D points is traced out in 3D space. Like the volumetric displays, the swept volume systems actually display true 3D points, within the physical size of the system. Solid Felix is also a "true" 3D display, where a solid crystal is doped with photonic elements in 3d space. The flourescent elements are excited by a laser to selectively illuminate points in 3D space. Finally, electronic holographic displays are a very new technology and are analogous to photographic holograms, except that the fringe pattern is created digitally rather than photographically. One day this will perhaps be the ultimate AS display, but computational costs and limited display resolution currently confine the practical realization of this technology to a very small scale.

The goal of this systems survey is not to show why one approach is better than another. Each system or group of systems solves some problems and ignores or creates other problems. For example, spatially multiplexed systems primarily suffer from a loss of resolution. Time multiplexed systems require much faster displays and refresh rates. Volumetric systems are limited by their physical size and lack of occlusion. Holographic displays are limited by current hardware constraints. Additionally, tracked systems are currently only single user and multi-view untracked systems display unnecessary viewpoints where no viewer may currently be located. To summarize the systems introduced above, Table I lists their salient features. Once common theme is that systems generally cannot change styles or operating modes; once a system is designed to operate a certain way, that is the way it must stay.

TABLE I

AS SYSTEMS COMPARISON

System name	References	Autostereo Method	Display Technology	Multiplexing Method	Number of viewers and views	Tracked
Varrier	Sandin et al. 2001 Sandin et al. 2005	static parallax barrier on printed film	LCD	spatial only	1 viewer 1 view per eye	yes
NYU	Perlin et al. 2000 Perlin et al. 2001	dynamic parallax barrier on Pi-cell	projector	spatial and temporal	1 viewer 1 view per eye	yes
Cambridge	Dodgson et al. 2000 Dodgson 2002 Cossairt et al. 2004	ferroelectric shutter synchronized w/ display	3 DMD projectors and spherical mirror	spatial and temporal	2 viewers 15 views per eye	no
DepthCube	Sullivan 2004	volumetric, 20 LC scattering shutters	fast projector	other, volumetric	many viewers infinite views	no
SynthaGram	Lipton & Feldman 2002	lenticular	LCD	spatial only	many viewers 9 views per eye	no
Sharp	Woodgate et al. 2000 Montgomery et al. 2001 Mather et al. 2003	dynamic rear (backlight) parallax barrier, switchable 2d/3d	LCD	spatial only	1 viewer 1 view per eye	no
Ren et al.	Ren et al. 2003	lenticular	LCD	spatial only	multiple viewers multiple views per eye	no
Jung et al.	Jung et al. 2003	lenticular	LCD	spatial and temporal	multiple viewers multiple views per eye	no
4D -Vision	Schmidt & Grasnick 2002 Relke & Riemann 2003	wavelength selective filter array	lcd in2002, 2003 projection screen in 2004	spatial only	multiple viewers 40 views per eye	no
Kleinberger et al.	Kleinberger et al. 2003	polarization filter	LCD or CRT	spatial only	1 viewer 1 view per eye	yes
Perspecta	Favalora & Lewis 2003	swept volumetric, rotating screen	projector	other, volumetric	multiple viewers infinite views	no
Miyazaki et al.	Miyazaki et al. 2003	swept volumetric, rotating mirror	LCD	other, volumetric	multiple viewers infinite views	no
Solid Felix	Langhans et al. 2003	static volumetric, 3d solid photoreceptive crystal	laser-excited crystal doped with florescent material	other, volumetric	multiple viewers infinite views	no
Holographic	Lucente and Galyean 1995	fringe pattern illuminated by coherent light	laser, LCD	spatial only	multiple viewers infinite views	no

2.7 AS rendering algorithms

In most AS implementations, the time complexity of generating 2 or more views is proportional to the number of views, or in big-Oh notation, O(number of views). There are some exceptions that will be noted, but this rule holds true in the majority of cases. It also makes sense; to render more views should require more time, typically a proportional increase. However, this is can be a problem for AS when rendering more views or higher quality forces performance to drop below interactive, real-time levels. The problem is especially pronounced in scientific visualization, where performance is strained by rendering large, complex datasets comprising tens of megapixels of screen resolution.

A review of some of the previous systems with regard to rendering algorithms and performance levels demonstrates this dilemma. In the Varrier system, a rendering loop draws one view followed by the other for each frame. If more views are to be drawn, the number of iterations of the loop are increased; clearly O(number of views). On modern graphics hardware, Varrier can render scenes of moderate complexity, approximately 20,000 polygons, at interactive frame rates of 30 Hz in its faster, lower quality mode. Doubling the number of views cuts this frame rate in half, which is barely acceptable by most definitions of interactivity. Without real-time interactivity, VR fails. To compound the problem, this performance is the case for the simplest Varrier algorithm (Sandin et al., 2005). Higher quality rendering requires multiple passes through the scene for each view, requiring more loop iterations.

Other space-only multiplexing systems display similar restrictions. The SynthaGram uses an algorithm to interleave pre-computed images, resulting in a 24-bit bitmap of the final image; also not interactive in real-time. The 4D-Vision system, like Varrier, loops over the number of views. The performance problem exists for the time/space multiplexing systems, as they need to multiplex views over time by their very nature. Many space-time systems use very fast CRTs or fast projectors to mitigate the problem, but are still limited by the same

factors: scene complexity and the number of views. For example, the Cambridge system displays a short pre-rendered animated sequence but is not interactive.

One group of systems is an exception to this rule. The common denominator in this group is the use of multiple image capture and/or display devices in parallel. For example, some integral imaging systems use multiple cameras in parallel with multiple display devices (Matusik and Pfister, 2004). The Kodak system (Cobb et al., 2003) uses two separate imaging engines that are focused separately to each eye. Hitachi (Kaneko et al., 2003) uses a similar idea but with 2 miniature projectors, and the FLOATS system (Kakeya et al., 2004) uses 16 projectors in parallel. The disadvantages of using parallel devices are increased cost and increased physical size of the system. For Dynallax, it is desired to use standard graphics workstation hardware as much as possible, without duplication of devices for view generation and display. Moreover, the use of parallel devices fixes the number of views displayed during design and construction of the system. One of the objectives of Dynallax is flexibility in terms of operating modes and number of views. This is impossible with a fixed number of parallel display devices.

Other problems in AS rendering relate to image quality. Latency and ghosting are common problems. Color distortion is another. Lack of image resolution, limited number of views or viewers, transitions between views (page-flipping), limited depth resolution, lack of occlusion, lack of ocular accommodation; every system is subject to at least one of these, if not more. Likewise, Dynallax cannot solve all of the problems facing AS and also needs to compromise and trade-off features of relative importance. However, with a dynamic parallax barrier, Dynallax provides a flexible solution that is configurable to optimize those qualities that are most important to the current application and mode of usage.

2.8 Optimizations

Some optimizations are possible to improve performance. Coherence is one such area where optimizations can be found. There are three types of coherence that can be exploited: temporal coherence, spatial coherence,

and perspective coherence. Temporal or frame coherence is the principle that successive rendered frames resemble each other. Therefore, computation can occur at lower temporal resolution and those frames that are not fully computed can be estimated from previous frames. Badt applied temporal coherence to ray tracing (Badt, 1988) More recently, the Talisman architecture utilizes temporal coherence by compositing image layers together, where an image layer is a logical image unit, not an entire frame. (Torborg and Kajiya, 1996) Spatial coherence on the other hand is the idea that changes usually occur slowly over distance, both in 3D world space and 2D image space. This is the key to level-of detail (LOD) division, that objects far away have perceptually low spatial frequency, and can be approximated by lower spatial resolution. Ezell and Hodges specifically apply spatial coherence to stereoscopic rendering in (Ezell and Hodges, 1990).

Finally, perspective coherence is the idea that stereo or multi-view perspectives resemble each other, and all views need not be computed from the start. This was seen earlier in (Adelson et al., 1991) with some low-level optimizations. More generally however, the idea extends to computing one or a few perspectives and performing affine transformations (warping) to produce the remainder of the views. This is the image-based rendering model, exemplified by QuickTime (Chen, 1995), and used more specifically to render stereo views of vegetation in (Borse and McAllister, 2002). Halle not only uses perspective coherence for optimization, but he bases an entire algorithm on perspective coherence (Halle, 1998). There, perspectives from multiple viewpoints are computed by transforming the problem from camera image space to epipolar plane image space, interpolating epipolar tracks of objects between two viewpoints to construct the intermediate views, and then transforming the results back to camera image space.

Optimization can also be achieved through architecture-based methods such as hardware acceleration and distributed computing. For example, Igehy designed a parallel graphics interface in (Igehy, 1998), and Woop et al. present a hardware ray tracer in (Woop et al., 2005). The recent rise of the GPU as a general purpose graphics processor is seen in the popularity of shader languages such as NVIDIA's CG language (Fernando and Kilgard, 2003) and the OpenGL shader language (Rost, 2004). Moreover, the modern GPU allows not only the design of custom shaders, but is taking over or supplementing higher-level algorithms that used to be dedicated

to the CPU, such as subdivision algorithms (Shiue et al., 2005), trimming and tessellation (Guthe et al., 2005), and ray tracing (Purcell et al., 2002). Of course design patterns in distributed processing also apply to computer graphics, namely parallelism through replication of hardware, for example performing ray tracing on a multiprocessor architecture. (Green and Paddon, 1989) It is not uncommon to see multiple GPUs, analogous to multiple CPUs. Moving outward, graphics can also be distributed among several machines forming a small cluster or a large supercomputer. This is the strategy in the Holodeck, a clustered parallel ray tracing system (Ward and Simmons, 1999). Dynallax relies both on hardware accelerations and distributed processing in order to achieve desired performance goals.

2.9 Light fields

One class of rendering algorithms pertinent to the topic of AS is the light field or lumigraph. Originally termed the plenoptic function, (Adelson and Bergen, 1991) a light field or lumigraph was first defined by Adelson and Bergen as a 7-variable function of eye position, spherical coordinates on a unit sphere centered at the eye, wavelength, and time. (a Cartesian coordinate version is also 7D) It essentially describes the intensity of all possible light rays entering the eye from all possible directions, of all wavelengths, at all times. In 1996, Levoy and Hanrahan proposed a 4D version of the plenoptic function called the light field (Levoy and Hanrahan, 1996), while Gortler et al. simultaneously termed it the lumigraph. (Gortler et al., 1996) In both cases, the number of variables was reduced to 4, namely intensity was mapped as a function of pixel location and ray direction. The technique allows all viewpoints of a non-occluding scene to be pre-computed and stored. The cost in terms of time and storage space is high, and pre-computation is done off-line. The benefit is real-time display for any viewpoint, given that the scene remains static and the light field already exists. This is ideal for some applications, for example architectural walkthroughs. The light field concept was subsequently implemented in hardware by Regan et al., although dimensionality was further reduced to a 3D light field by considering only horizontal parallax. (Regan et al., 1999) More recently, the light field concept emerged in the PixelView

architecture (Stewart et al., 2004) using a 4D light field function that provided both vertical and horizontal parallax.

An interesting comparison can be made between traditional monoscopic rendering vs. light fields on the one hand, and tracked AS vs. untracked multi-view AS on the other. In a sense, monoscopic rendering and light fields are algorithmically opposite, as tracked and untracked AS are systemically opposite. Traditional rendering computes only the desired viewpoint at any one time, looping as necessary to produce multiple viewpoints, but only computing those views that are presently required. This is also the case with tracked AS. Light fields, in contrast, contain information about all viewpoints, as with untracked multi-view panoramic AS. Each approach has pros and cons. The light field / untracked AS methods strive to pre-compute all possible information, based on the assumption that it cannot be computed in real-time. The advantage is that views are ready when needed, as in a walkthrough type application. The disadvantage is that much computation is performed that will never be used, as a user will only be located at a small percentage of all of the pre-computed views. The other disadvantage is that scene content cannot change dynamically in real time. The traditional rendering / tracked AS methods strive to compute data just in time, only as needed. The drawback is that it may not be possible to keep up with changing viewpoints, but the advantage is that all computational resources can be channeled to only the required viewpoints.

3. SOLID STATE DYNAMIC PARALLAX BARRIER

Chapter 3 explains in more detail why a dynamic parallax barrier is desired, how one is constructed from solid state LCD components, and the relative merits of employing a dual stacked LCD monitor for the Dynallax display technology. First, Section 3.1 motivates the problem with a review of the drawbacks of a static barrier and one naive solution using a mechanical barrier. Section 3.2 introduces the solid state LCD barrier and illustrates its advantages over a mechanical system. The intricacies of LCD technology may not be familiar to the typical reader, so a short background review of LCD hardware concepts is included in Section 3.3 before extending those ideas to a stacked 2-layer display in Section 3.4. Finally, the chapter concludes with a summary of the benefits of a stacked display for parallax barrier AS.

3.1 Motivation

As explained earlier, a static parallax barrier imposes limitations on the resulting AS system because the system parameters are fixed during the construction of the barrier. Hence, overall system operation is dictated by how the barrier is designed and built. A few barrier parameters such as period, tilt angle, and duty cycle affect gross overall characteristics such as number of views and viewers, brightness and contrast, level of ghost, working range, and a number of other features. If the barrier can be varied dynamically, a multi-modal system is possible. Operation could be mono, stereo or multi-view, tracked or untracked, single or multiple viewers. Moreover, basic AS relationships can be studied and quantified because experimentation is easy, and completely new barrier types and patterns can be investigated.

To further motivate the discussion, let us consider a mechanical approach to a variable barrier first. This is not the method employed in Dynallax; it is more of a thought experiment of how a dynamic barrier can be constructed and introduces the concept in practical terms. Imagine several layers of thin barriers stacked in front of the display, with the ability to slide the barriers with respect to each other. See Figure 3. By moving the barriers with respect to each other or removing certain ones, a variety of barrier periods and even new barrier

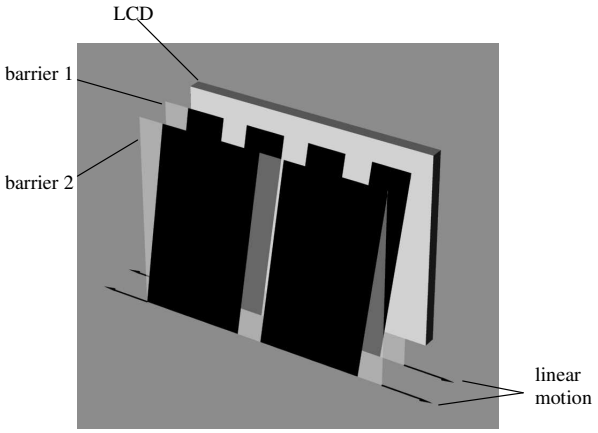


Figure 3. Mechanical dynamic barrier

patterns can result. The barriers can be moved with a servo-mechanical drive system and suitable controller to produce the desired period.

The disadvantages of this system are the limited combination of resulting barriers and the mechanical complexity of the drive mechanism. This complexity can present itself in cost, size, weight, maintenance, etc. It should be noted that in general, AS systems are not always strictly solid state and some do contain mechanical components. Some examples are (Favalora and Lewis, 2003), (Miyazaki et al., 2003), and (Kakeya, 2003). The other disadvantage of this method is the limited capability to produce output barriers, since the output is the result of a fixed set of static input barriers. On the other hand, such a system exhibits some positive characteristics such as high accuracy, high contrast, and low light loss. Overall however, a modern more simple solution can be obtained using a solid state dynamic barrier.

3.2 Solid state dynamic barrier

The solid state approach is to stack 2 LCD displays, one in front of the other, and generate the dynamic barrier on front display while the scene image is drawn on the rear display, as in Figure 4. By rendering a pattern

of black and white stripes on the front LCD and stacking the two LCDs such that the light from the rear display passes through the front display before reaching the eye, the front LCD acts exactly as the static barrier did.

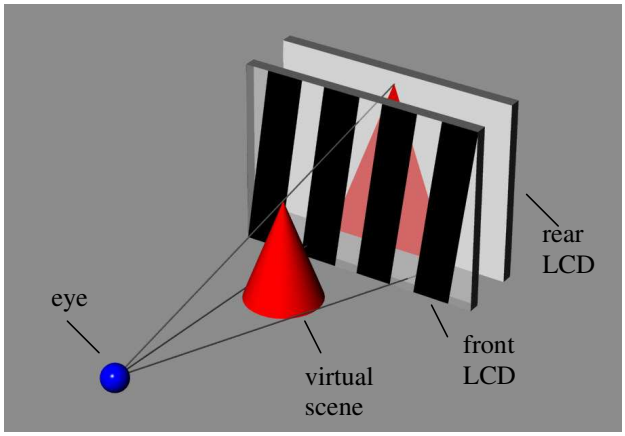


Figure 4. Solid state dynamic barrier

The advantage is that barriers of varying periods can be rendered in real time. However, barrier patterns need not be limited to strips as in the mechanical approach, nor need they be binary. Colored wavelength-selectable barriers, Poisson dot distributions, or any other imaginable patterns can be experimented with. The system is inexpensive, has no moving parts, and is controllable by conventional computer graphics software and hardware. The disadvantages compared to the static or mechanical barrier are reduced contrast, more light loss, and light leaks.

3.3 LCD technology primer

To better understand the function of the stacked LCD and how it is used for parallax barrier AS, some relevant background material is presented in this section about LCD technology.

LCD Construction and operation

Figure 5 shows the composition of layers that comprise a typical single layer LCD display.

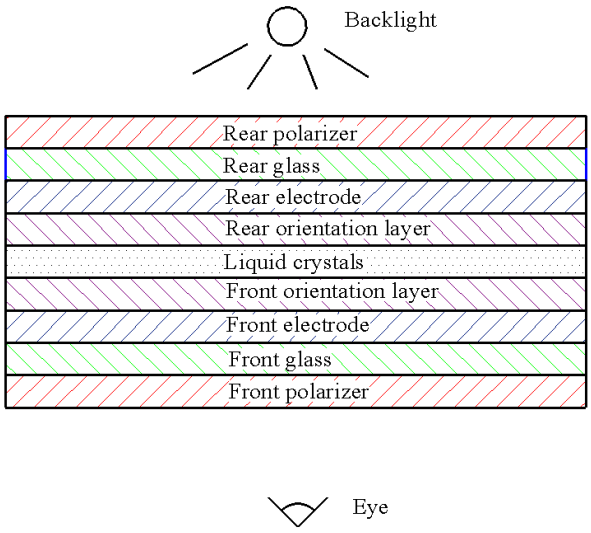


Figure 5. LCD sandwich

Usually located behind the back polarizer is a light source, commonly called a backlight. It is important to recognize that an LCD display is a transmissive medium. The liquid crystals (LCs) do not emit light; they only selectively transmit light provided by the backlight or other light source. In a twisted nematic (TN) LCD, the two orientation layers and two polarizer layers are oriented 90° apart as shown in Figure 6. (Wright, 2002)

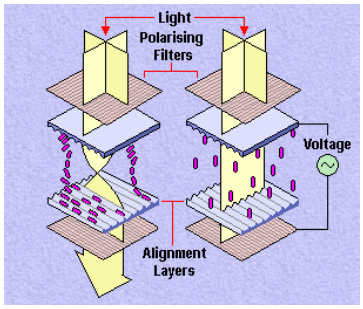


Figure 6 courtesy of (Wright, 2002)

Figure 6. Polarized light in a TN LCD

Since the two orientation layers are 90° apart, LCs in the absence of a voltage are forced to twist across the thickness of the LC layer to conform to the orientation layers. The polarizers are in the same directions as the orientation layers (also 90° apart), so light twists as it passes through the LCD (still under a no voltage condition). When voltage is applied, the LCs straighten, standing up 90°vertically, so that light is not twisted as it passes through the LC layer. Since the polarizers are orthogonal to each other, no light passes through the front polarizer and the display is black. Super twisted nematic (STN) LCD displays also exist that have higher degrees of twist, up to 180° or even 270°, but they are not very common.

Other LCD terminology refers to the method of lighting the display and controlling the LC elements. The transmissive LCD is most common, where the light source is located behind the display (backlight) and light travels from back to front. Other variations are the reflective LCD, where light originates in front of the display (which can also be ambient light), is reflected from a mirror in the back of the display, and then travels from back to front as before. The transflective LCD is a hybrid of the transmissive and reflective methods that includes a half-silvered mirror to use both reflected ambient light and transmitted backlight to illuminate the display. These are attempts to mitigate the poor performance of LCD displays outdoors or in bright environments. To control the LC elements, the most common method today is the thin film transistor (TFT) active matrix, which includes an individual transistor to regulate the voltage to each pixel or RGB sub-pixel. Dynallax employs TN, TFT, transmissive LCD technology for both the front and back displays.

3.4 Stacked LCD construction

Two LCDs can be stacked together if the backlight of the front layer is removed and the two LCDs are oriented orthogonal to each other. Each individual LCD has the construction of Figure 5, but both are mounted integrally in the same housing and share the same backlight. The light is twisted twice, once in each layer, and needs to be re-polarized between layers. The idea is illustrated in Figure 7.

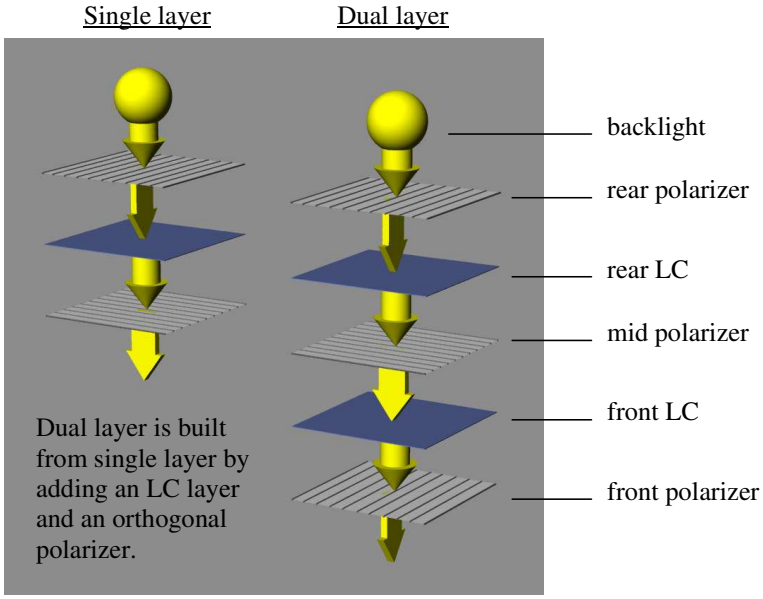


Figure 7. Stacked LCD construction

The intensity of the final output is the product of the intensities of the individual LC layers:

$$I_{final} = I_{rear} * I_{front} \qquad (1)$$

A company called Puredepth in California produces a model MLD3000 2-layer 17" LCD display with 1280x1024 resolution in each of 2 layers. (PureDepth, 2005) The product is commercially available and retails at the time of this writing for approximately $1800, and this is the display used for the prototype Dynallax system. The target applications for the product appear to be semi-transparent stacked user interfaces as a new user interface paradigm, but obviously the display is used here for an entirely different purpose. Other relevant specifications are 18 bit color, 7mm physical separation between the two layers, 250:1 contrast when viewer on-center, which diminishes to 80:1 when viewed at a 45° angle off-center. The overall display brightness is 100 cd/m^2. In general, the contrast is less than a single layer display, and the brightness is lower as well. For example, a comparable single layer display has 350:1 contrast and over 200 cd/m^2 brightness (NEC, 2005).

NuerOK Optics (California) debuted a stacked LCD display at the SPIE 2002 Stereoscopic Displays and Applications conference (SPIE, 2002). In 2005, a three-layer stacked display became available from NuerOK Optics (Loukianitsa et al., 2005), although the authors use the display in a different way. Rather than generating a dynamic parallax barrier, they attempt to simulate a 3D light field by the sum of light attenuated by the LC layers along each ray direction. It is unclear whether any advantage would be gained from higher numbers of layers for the Dynallax barrier method, and whether the loss of brightness would be justified by the additional layer(s). Loukianitsa et al. cite that four or more layers are not justified by the light loss in their system.

3.5 Features and advantages for parallax barrier AS

Aside from being able to support multiple viewers, using a dynamic parallax barrier constructed from a fully addressable LCD stacked display offers many advantages. Some of these features are beneficial for the two-view tracked mode of operation, while others are desirable for other modes of operation and several others provide paths for future AS research. Some key benefits are:

- Ability to adjust barrier angular orientation (line tilt) easily and to find angles that minimize Moiré effects
- Expanded minimum view distance range
- Elimination of experimental registration of physical barrier pitch with virtual barrier pitch
- Ability to produce different types of barriers such as grayscale or barriers consisting of shapes other than lines. Although this feature is not further pursued within this research, it is a possible future direction.
- Ability to adjust duty cycle to adjust size of channels and guard bands
- Ability to turn barrier on/off and selectively work in monoscopic or AS modes (2D or 3D)
- Ability to generate other AS methods such as non-tracked multi-view panoramagrams. This feature is also not developed further here, but is left for future study.
- Reduced color shifts, eliminating the need for 3 passes to produce sub-pixel accuracy

A number of these features are explored and quantified in detail in Chapters 4 and 5.

4. IMPLEMENTATION

The realization of the Dynallax method into an actual working prototype system is the subject of this chapter. Section 4.1 describes the overall system structure including both the computation sub-system as well as the display sub-system. The next two sections define the software structure: Section 4.2 covers the main Dynallax software modules for producing and controlling the dynamic barrier AS while Section 4.3 demonstrates auxiliary components that are required to form a complete system. Lastly, Section 4.4 explains how such a system is evaluated through the use of test patterns.

4.1 System structure

Sub-section 4.1.1 focuses on the machines that comprise a 3-node mini-cluster, as well as a single stand-alone configuration. Details include overall structure, division of labor, and hardware specifications. Sub-section 4.1.2 includes the specifications for the stacked displays used in the research. A stock display as well as a custom-built unit are examined.

4.1.1 Computation

The system is composed of a 3-node computational cluster, a dual stacked LCD monitor, and appropriate rendering and control software. These components are pictured in Figure 8. The computational nodes comprise a small 3-node mini-cluster and each of the nodes has the specifications given in Table II.

The organization of the cluster is as follows: The master node serves to control the overall work flow while the first slave controls the rendering of the dynamic barrier, the front layer of the LCD display. The second slave renders the multiplexed scene imagery on the rear LCD layer.

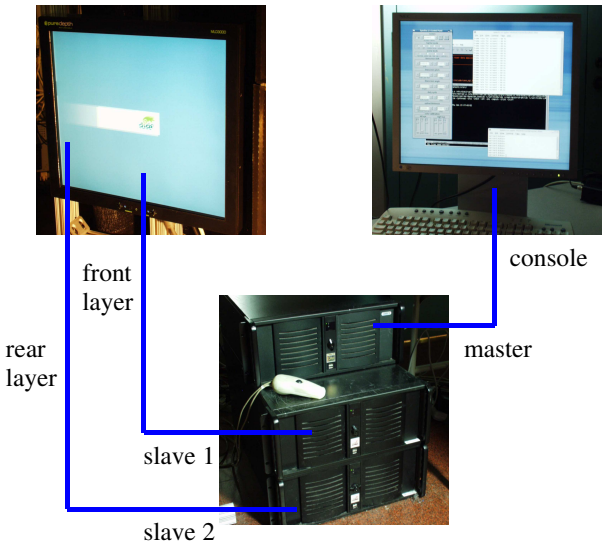


Figure 8. Dynallax mini-cluster structure

TABLE II

CLUSTER NODE SPECIFICATIONS

Number of processors	2
Processor type	Intel Xeon
Clock rate	2.4 GHz
cache	512 KB
Network adapter	1 Gbps Ethernet
memory	1 GB
Graphics adapter	NVIDIA Quadro FX3000
Graphics bus	AGP
Operating system	Suse Linux 10.0

A more compact node structure that is amenable to scaling up to a tiled display condenses all three processes onto a single machine. Logically, the work flow is identical to that above, in that three MPI processes remain, except that they reside on the same machine. The ability to partition processes as desired among machines, including multiple processes on a single machine, is one of the strengths of MPI. Graphics hardware, the bottleneck in AS rendering, is extended with a second graphics card. The first GPU is shared between a console and the front screen of the stacked display, and the second is dedicated to the rear screen of the stacked display. Note that a console is optional; Dynallax can be launched and controlled from any remote machine. NVIDIA GeForce 7900 GPUs (Nvidia, 2006) comprise the two graphics cards. Table III lists the specifications of the improved stand-alone computer.

TABLE III

STAND-ALONE SPECIFICATIONS

Number of processors	2
Processor type	AMD dual core 64 bit Opteron
Clock rate	2.0 GHz
cache	1 MB per core
Network adapter	1 Gbps Ethernet
memory	4 GB
Graphics adapter	(2) NVIDIA GeForce 7900
Graphics bus	PCI-express
Operating system	Suse Linux 10.1

4.1.2 Display

Two display models were tested during this research. The first is a stock dual stacked LCD display available from PureDepth in California (PureDepth, 2006). Later, a custom display consisting of two dissimilar LC panels was designed and constructed by the author in order to investigate various aspects of the stacked display as it

pertains specifically to Dynallax. This section briefly describes both display versions, while the results of extensive tests performed on both models are presented in the following chapter.

The dual stacked LCD display is the PureDepth MLD3000. Its specifications are given in Table IV, and Figure 9 depicts a user interacting with the complete prototype display system.

TABLE IV

STOCK STACKED DISPLAY SPECIFICATIONS

Manufacturer	PureDepth
Model number	MLD 3000
Size	17 inch
Resolution	1280x1024
Dot pitch	.25 mm
Layer separation	7 mm
Contrast on-center	250:1
Contrast off-center	80:1 @ 45 degrees
Brightness	100 cd / m^2
Color depth	18 bit

Figure 9. Prototype Dynallax system

The custom display consists of two dissimilar LC panels. The rear panel is a common color display, in this case a 20-inch model from NEC. However, in order to physically accommodate a second panel stacked in front of this display, the rear unit is removed from its plastic housing, and its sheet metal inner enclosure is modified by cutting a portion of it away. The power supply, which was located in this area, is re-located.

Then, the front LC panel is taken from a second stock LCD monitor, but this monitor is an expensive, monochrome medical-grade imaging device that has high resolution and contrast. The retail price of this monitor is on the order of $5000, and ultimately all that is needed from it is the LC panel, driver circuit board, and power supply. After carefully disassembling the monitor, these items are removed and mounted onto the rear panel. The reasons for selecting a monochrome front panel are explained in Chapter 5. As explained in Chapter 3, the angles of polarization in the layers of a dual stacked display must be orthogonal. Rather than attempting to remove and re-orient the polarizing element, the most direct approach is to simply position the entire top panel 90 degrees with respect to the lower. Conveniently, this orientation also creates physical space for the narrow circuit boards that connect the horizontal rows of the active matrix to the LC panel. Other boards of the front panel such as the driver board and power supply are located wherever convenient, constrained mainly by cable length. Figure 10 shows various photographs of the custom display, and Table V specifies the main parameters of both the rear and front panels of this unit.

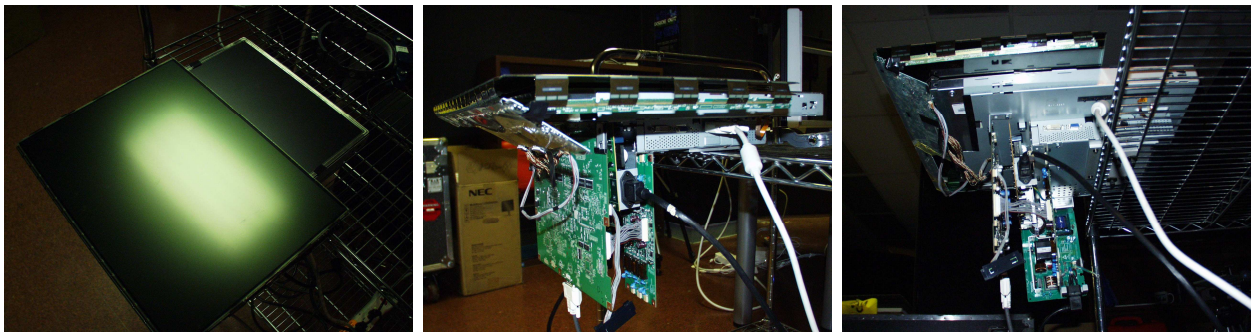


Figure 10. Views of custom stacked display

TABLE V

CUSTOM STACKED DISPLAY SPECIFICATIONS

	Front Screen	***Rear Screen***
Manufacturer	NEC	NEC
Model number	MD21GS-3MP-BK-CB	2080UX+
Size	21.3 inch	20.1 inch
Resolution	2048x1536	1600x1200
Dot pitch	.21 mm	.25 mm
Layer separation	5 mm	N/A
Contrast off-center	700:1	400:1
Brightness	700 cd/m^2	250 cd / m^2
Bit depth	10 bit	24 bit

4.2 Dynallax software

Dynallax 1.0 is a custom software package written by the author. It began as an outgrowth of the Varrier3.0 library, also written by the author, that controls the Varrier AS system, but eventually diverged. Varrier is based upon the CAVElib™ architecture (VRCO, 2006) while Dynallax is not. Like Varrier, Dynallax includes provisions for inter-cluster communication via the "message passing interface" or MPICH (MPICH, 2005). The Varrier method originally used the depth buffer to spatially multiplex two perspective views into each rendered frame by rendering into the depth buffer a virtual replica of the physical barrier. (Sandin et al., 2001, 2005)

In more recent implementations, Varrier contains a shader-based GPU implementation of an algorithm that does not depend on the depth buffer, called Varrier Combiner (Kooima et al., 2007). The shader-based algorithm used in Dynallax is similar to and based on Varrier Combiner, and expands that algorithm to incorporate 4

channels instead of two, to render both the visible front barrier as well as to modulate the images of the rear screen, and to include two barrier patterns (single period and dual period) as explained in Chapter 5.

The Dynallax software is divided into three modules: master module, slave module, and controller module. Each of these is explained in sub-sections 4.2.2, 4.2.3, and 4.2.4, respectively, following an overview of the software structure in sub-section 4.2.1. This section closes with a description of the user interface for controlling the Dynallax system in sub-section 4.2.5.

4.2.1 Software structure

Dynallax can be conceptually organized into three modules in terms of the functions performed: master, slave, and controller. This is a different view of the system than the process view taken earlier. To map one view to the other, the master and controller are two modules within the master process, while the slave module is replicated in each of the two slave processes. The slave module performs slightly different tasks depending on whether it is in the front screen slave or the rear screen slave. The following sub-sections elaborate on the function of each module.

4.2.2 Master module

The master module runs the control panel by which the user can operate Dynallax, and based on user input the master sends messages to the slaves every frame. The messages contain three kinds of data: barrier data, head tracking position data, and application content data. In order to accomplish this, the master manages the receipt of head tracker and wand data from a tracking daemon. Initial barrier data and system data is read from a Lua configuration file, and further changes to the barrier are performed as directed by the user's interaction with the control panel and by the controller module.

4.2.3 Slave module

The slave module has a different responsibility depending on whether it is the front or rear screen. The front screen runs the front shader program in order to render a visible parallax barrier. The parameters of the barrier such as period and shift are contained in a message received from the master prior to each frame. In the case of monoscopic 2D mode, the front slave simply renders a white screen that is transparent. The rear slave performs a sequence of steps for each frame. It renders each of the eye perspectives into an off-screen buffer, and then modulates these images together with a perspective-corrected model of the projection of the front barrier onto the rear screen. In the case of monoscopic 2D mode, the rear slave simply renders the first eye perspective and nothing more is done. Each of the front and rear screen slaves sends a ready message to the master after the rendered frame has been completed, indicating that each is ready to receive more data.

4.2.4 Controller module

Three basic operating features are supported by the controller: the maintenance of optimal barrier period due to viewer's distance from the screen, the rapid steering of output channels to keep pace with rapid viewer head movements, and the maintenance of optimal barrier period when two viewers are present. Some of these functions operate within the same controller cycle while others are mutually exclusive, depending on events triggered by the viewer. The controller is called upon for each frame to adjust the following barrier parameters: barrier period, duty cycle, and shift, given the current barrier parameters and user(s) head position(s). The flow chart in Figure 11 illustrates controller behavior. The view distance block sets the optimal barrier period based on the viewer's distance from the screen. The rapid steering block sets the barrier shift based on rapid head movements, and the 2 viewer block sets the barrier period based on viewers' head positions when two viewers are present. Details of the computations performed by each block are explained in Chapter 5.

The view distance function occurs during every cycle, and controls the barrier period in order to maintain optimal channel separation based on the viewer's distance to the screen. If two viewers are present, their average view distance is used in order to compute barrier period. The details of the barrier period calculation due to view distance are given in Chapter 5.

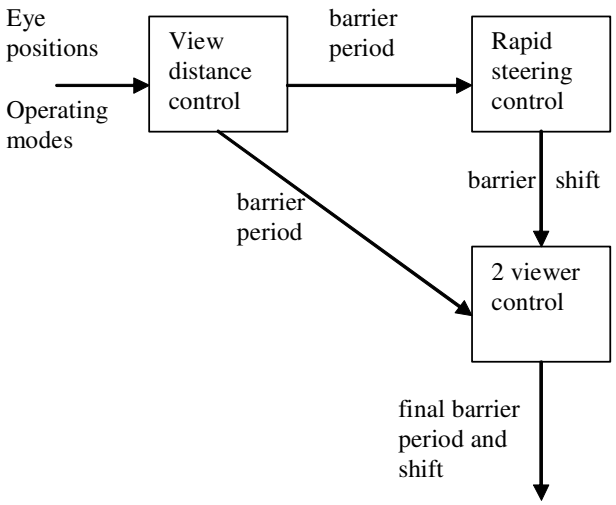


Figure 11. Controller module structure

The rapid steering function adjusts barrier shift to steer channels to the viewer more quickly than the channels can actually be updated. This feature cannot operate at the same time as two-viewer mode, so rapid steering is automatically disabled whenever two-viewer mode is enabled. Also, rapid steering can be manually disabled. This is useful for running tests and for calibration; for example barrier shift cannot be manually calibrated while the controller is constantly modifying that parameter in rapid steering mode.

When two viewers are present and two-viewer mode is enabled, the barrier period and duty cycle need to be further modified. The period needs to be at least twice as large as compared to single viewer mode, although

usually it needs to be expanded even further to avoid conflicts. The two-viewer function searches for a barrier period that is optimal, based on the head positions of the viewers. Details of this computation are given in Chapter 5.

4.2.5 User interface

Dynallax includes a control panel for managing its features and permitting easy experimentation with barrier parameters and modes. As seen in Figure 12, controls exist for the following categories:

- barrier modes such as pixel or sub-pixel, 3D or 2D
- scene modes such as normal or calibration patterns
- barrier parameters such as period or pitch, angle, shift, and optical thickness
- operating modes such as single or 2 viewer
- calibration controls such as test pattern colors

Figure 12. Dynallax control panel

4.3 Auxiliary software components

Before creating and controlling a dynamic barrier and using it to modulate VR scene content, a great deal of perhaps less interesting work needs to happen. However, finding practical solutions to these routine "housekeeping" software operations can help make a software system that is small, compact, easy to understand and modify. Three of these software engineering topics are covered here: the management of graphics window context, creating and maintaining system configuration files, and inter-process communication. In all three cases existing solutions are drawn upon and are documented in sub-sections 4.3.1,4.3.2, and 4.3.3, respectively.

4.3.1 Window management

A full-size window with an openGL rendering context is created and managed on each of the front and rear screens of the stacked display using a software API called SDL, or Simple Directmedia Layer (SDL, 2006). SDL is similar in operation to GLUT (GLUT, 2006), another popular window management library, although SDL is a more current API that continues to be maintained. In previous AS implementations such as Varrier, this functionality was performed by CAVELib (VRCO, 2006), as well as in the early stages of Dynallax research. SDL has since replaced the window management features in Dynallax and other tasks performed by CAVELib such as view frustum are now performed at a lower level in openGL. SDL also has functionality for handling input events such as keyboard and mouse input, although Dynallax handles events with FLTK, or Fast Light Tool Kit (FLTK, 2006)

4.3.2 Configuration management

One of the mundane but necessary aspects of any large software project is configuration management. Configuration files contain system data that needs to be read and parsed when the system starts. In early

Dynallax versions, CAVELib managed some of this configuration data while other information was parsed within custom C++ functions of Dynallax code. Custom parsing of complex configuration files within application programs is seldom robust. A small custom "language" is actually invented every time a configuration file is developed, and language parsing is not a trivial task. Usually only minimal error checking is performed because robust error checking is difficult and tedious.

A better solution and one employed in later Dynallax versions is to use a ready-made scripting or configuration language to write configuration files and allow the scripting language parse the data. In Dynallax the language used is Lua (Lua, 2006). A configuration file takes the form of a Lua program, which is a text file that can be read by either an interpreter or by an API of C++ functions. The latter case is employed in Dynallax, and data values are transferred between Lua and C++ through a virtual stack. The net result has three advantages over the manual parsing approach. The C++ code is shorter because much of the work is done by Lua; it is robust because error checking is done by Lua, and a common configuration language can be used for other projects rather than re-inventing new languages each time.

Dynallax uses two Lua configuration files. The first contains all of the system parameters such as screen corner locations, screen resolutions, and barrier parameters. In the past this data was contained in CAVELib configuration files as well as in a custom barrier configuration file. The second type of configuration information in Dynallax is application model data, containing models to be loaded and their positions and orientations within the VR scene.

4.3.3 Message passing

Three processes exist when Dynallax is running: a master process and two slave processes for front and rear rendering screens. Each process may run either on a separate machine or these processes can co-exist on the same machine. The key entity to consider in Dynallax's division of labor is the process, not the machine. The

API that handles all inter-process message passing is MPICH-2, or the **M**essage **P**assing **I**nterface **CH**ameleon version 2 (MPICH, 2005). This message passing library, developed at Argonne National Laboratory, supports point-to-point and broadcast synchronous and asynchronous communication efficiently.

Dynallax relies on both synchronous and asynchronous communication modes. A synchronous communication command *blocks* until the completion of the communication call; for example, a synchronous receive command will wait until a message is received. This can be useful at times to rendezvous processes, and may be undesired at other times. Asynchronous communication calls return to the calling routine immediately after being called. Interrupt-driven communication can be programmed using asynchronous message transmission.

The controller module in Dynallax (Section 4.3.4), together with user input, selects one of several modes of operation at any given time. When Dynallax is in rapid steering mode, communication needs to occur asynchronously so that the front and rear screens update at their own maximum possible rates. On the other hand, there are cases when screen synchronization and synchronous message passing is desired, such as when rapid steering is disabled or when two-viewer mode is enabled. Even with rapid steering mode enabled, occasional barrier period changes occur synchronously to minimize distracting visual artifacts. Therefore, when a barrier period change occurs, the slave processes rendezvous; the change is made using synchronous communication, and then the processes continue as they were.

Slave processes can also be synchronized with each other, even though there is no direct communication between them, with a *message barrier* (unrelated to a parallax barrier). In this context, a barrier is a logical entity that operates as follows. If all three processes reach a barrier command, then none will proceed beyond the barrier until all three have reached it. Whenever Dynallax switches to synchronous mode, a barrier is employed as a secondary method of synchronizing the two slaves with each other.

Communication in Dynallax is duplex: slaves send a “ready” message to the master, and upon receiving it the master sends new data to the slaves. This handshaking protocol permits slaves to run asynchronously when desired and yet to receive the most up-to-date information whenever they ask for it. Figure 13 illustrates a flow chart of how and when Dynallax employs each of the synchronous / asynchronous modes described above.

Note that in all but the most trivial cases, the rear screen is a slower process than the front screen. That is because it is computationally less expensive to render the front screen barrier than to render a complex, modulated scene in stereo on the rear screen. (This fact is what makes the rapid steering mode possible at all.) In synchronous mode, both screens run at the frame rate of the slower, rear screen because they are in lock-step with each other, frame for frame. In asynchronous mode, each screen runs at its fastest possible frame rate.

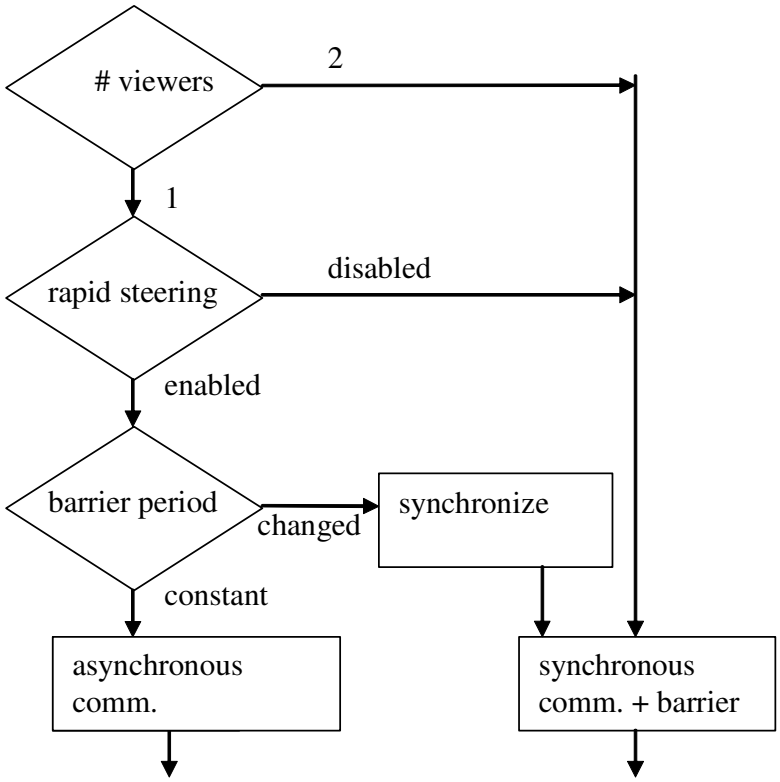


Figure 13. Synchronous and asynchronous communication modes

4.4 Calibration and test patterns

Testing and calibrating an AS system is performed with the aid of test patterns. As in (Sandin et al., 2005), the color and cross-bar patterns are used heavily in Dynallax, and a number of results in the following chapter rely on these. In the color pattern, a single large polygon of a different color is rendered for each eye; when the system is well-calibrated, only that color light should enter the eye. When the viewer is positioned at the correct COP and covers one eye, the other eye should see only the desired color light. For example, the left eye will see a solid display of green while the right eye sees a solid display of blue. When two viewers are present, each of the four channels is represented by a different color, or occasionally a color may be repeated for two viewers when there is no ambiguity. The cross-bar pattern is similar, except that vertical, horizontal, and angled bars of white and black represent the eye channels. For example, the left eye sees vertical bars while the right eye sees horizontal.

A first-person perspective for each eye channel results when the eye is positioned at the COP from which the test pattern is rendered. This is useful in many instances, but in other cases the researcher wishes to see a third-person or "birds-eye" view of the entire system output. This "omniscient" view results when viewing from infinity, or in a practical sense from much further away than the COPs from which channels are rendered. For example, if the viewer's position is set to 24 inches from the display and the color pattern is rendered, then the upper images in Figure 14 illustrate what the viewer sees in each eye when at the correct location, while the lower images shows what a viewer sees from a 120 inch view distance. From this far vantage point, both channels are visible in the same view, and they are separated by the interocular distance when the view distance becomes large enough.

Figure 15 explains why the "far test" works. A parallax barrier AS system focuses channels at the plane of the COPs, 24 inches away from the display in the above example. As an analogy, imagine a film of the same physical dimensions as the display positioned at that plane, capturing the images in space focused there. Next, imagine the film remaining stationary, but a person viewing that film from much farther back, such that the

entire film is visible with approximately parallel light rays from the film to the viewer. The result is the far away "birds-eye" view, where all channels are visible simultaneously, and channels are spaced by the interocular distance. This spacing can be physically measured by superimposing a scale or reference onto the display and comparing the channel spacing to the reference measure.

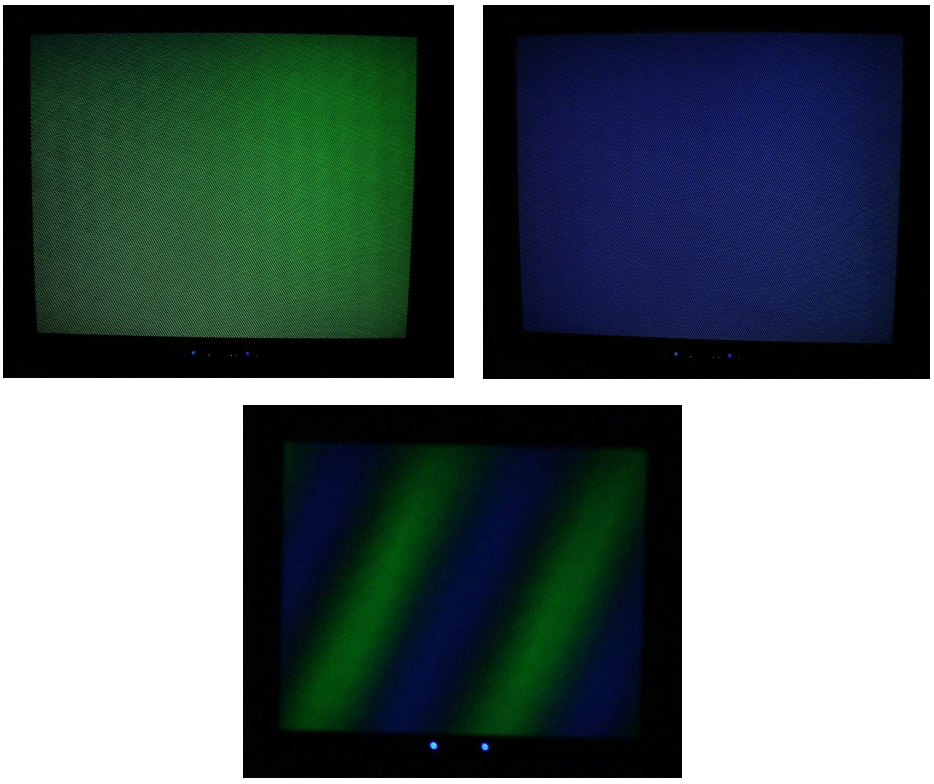


Figure 14. Test patterns viewed from correct distance (top) and from afar (bottom)

The fact that a well-calibrated parallax barrier system can focus hundreds of lines of spatially-multiplexed pixel or sub-pixel channels into a small number of coherent images at COPs remains one of the amazing characteristics of this type of AS system. The far test makes this point explicit, a result that is sometimes underestimated in the first-person perspective. Another often-overlooked fact is evident in the far test: channels repeat laterally outward in both directions. Not only are green and blue channels focused at the left and right eye

positions, but they repeat at “virtual lobes” infinitely far left and right, at diminishing intensity. This repetition will become a key point in the discussion of the two-viewer mode in Chapter 5.

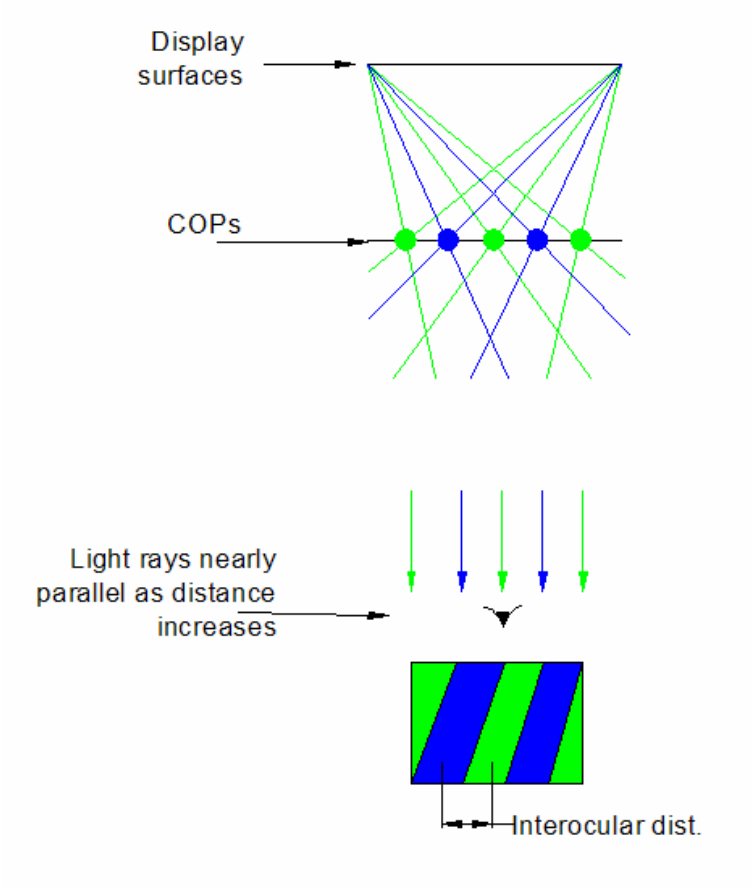


Figure 15. Viewing all channels simultaneously, spaced by the interocular distance, in the far test

5. RESULTS

Chapter 5 identifies, discusses, and analyzes the results discovered during the course of Dynallax research, documenting findings using reproducible methods such as screen shots, photographs, and quantifiable measures. Along with identification and documentation of a discovery, equations and other numerical models are developed that explain and predict the conclusion, further illustrating its validity and usability to predict similar conditions. In sum, this chapter presents a comprehensive model of dynamic barrier AS within the context of the Dynallax system.

Section 5.1 lists some of the barrier parameters that are easily varied through Dynallax. Next, Section 5.2 presents two convenient side effects that result from the use of a dynamic barrier. The subject matter becomes progressively more detailed as the reader moves beyond these first two sections through the remainder of the chapter. A mathematical definition of the sub-pixel continuous barrier model is presented in Section 5.3, including a justification for its existence compared to other discrete models. Section 5.4 covers the concept of viewer distance from the screen, and illustrates how Dynallax maintains optimality over a wide range of view distances. Section 5.5 explains the way that Dynallax mitigates parallax barrier sensitivity to system latency through the application of a rapid steering mode. The largest topic within the chapter is the solution of the two-viewer mode problem, presented in Section 5.6. The two-viewer problem demonstrates the need for better stacked display solutions, presented in Section 5.7. Finally, the chapter closes with a discussion of net effective resolution and a table quantifying this metric for various Dynallax conditions in Section 5.8.

5.1 Variable parameters

To begin, this section lists a number of barrier parameters that are easily variable through Dynallax. The first is illustrated in Sub-section 5.1.1, the fact that the barrier can be turned off, either in total or in part, converting the display from 3D to 2D and vice versa. The remaining subsections, 5.1.2 – 5.1.4, illustrate other aspects of the barrier that can be changed in real time, namely barrier period, barrier duty cycle, and barrier angle, respectively.

These barrier parameters form the basis for the rest of the Dynallax method and their dynamic adjustability is crucial to the operation of the system.

5.1.1 3D – 2D switchable display

The first result of Dynallax and one of the most useful is that the front barrier screen can be disabled, converting Dynallax to a 2D monoscopic display. For example, Figure 16 illustrates three uses of this feature. The upper left image in Figure 16 depicts 3D AS mode; the upper right image shows 2Dmono mode while the center bottom image demonstrates mixed-mode, both AS and mono content within the same display. Here, the Mars rover model is rendered in AS while the toolbar in the left column of the screen is rendered in 2D mono mode. Any 2D content is rendered in mono mode, and no barrier lines occlude the data.

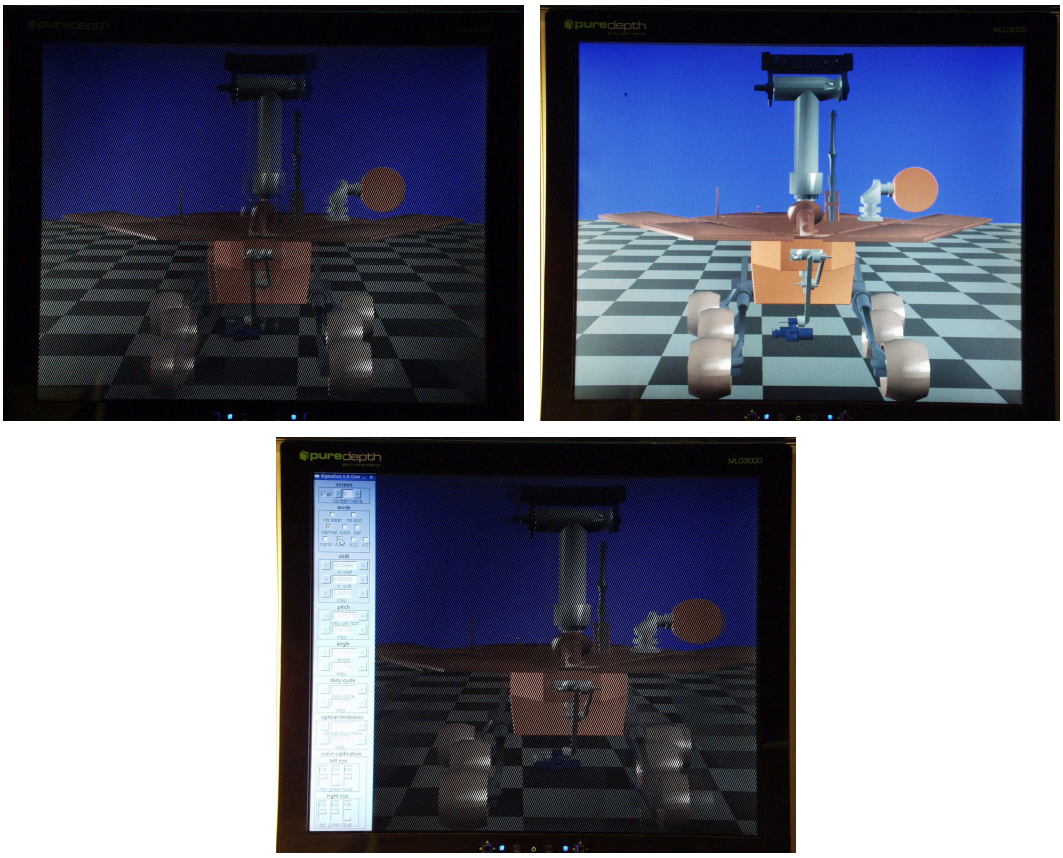


Figure 16. Switching between stereo, mono, and mixed modes

In previous static barrier AS systems such as Varrier, 2D content was possible to render, but readability was limited due to the presence of the physical barrier lines. Dynallax poses no such limitation. Moreover, 2D – 3D switchability is a desirable feature in commercial parallax barrier AS displays. This was a popular selling point for the Sharp display (Woodgate et al., 2000), and currently the trend is toward switchable 2D -3D lenticular displays. In particular, Phillips currently offers a 42 inch model based on blocks of 8x8 lenticular elements. (de Boer et al., 2007). Dynallax's dual stacked LCD construction offers 2D – 3D switchability on a per-pixel basis.

5.1.2 Barrier period adjustment

A key advantage of Dynallax is the ability to change the period of the physical barrier, allowing the system to accommodate one or two viewers, and also to expand the minimum view distance. Figure 17 shows some sample barrier periods of 310 lines per foot (lpf) on the left and 155 lpf on the right.

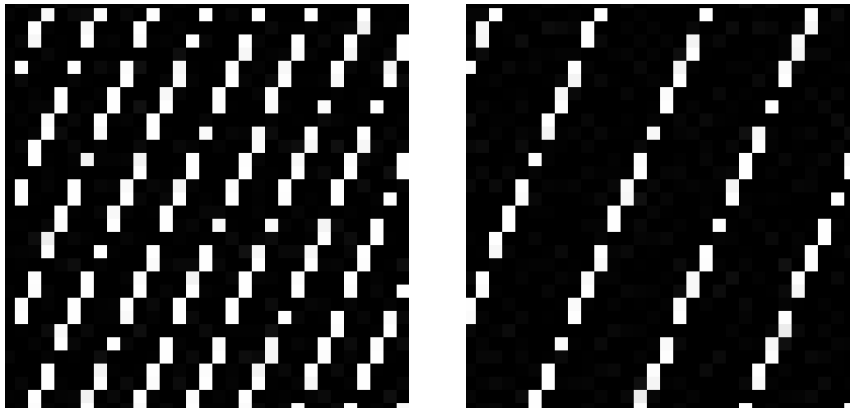


Figure 17. Barriers for single-viewer (left) and two-viewer (right) modes

One question that often arises concerns the discrete nature of the barrier shown in Figure 17, as opposed to the continuous physical that exists in static barrier systems. Specifically, the presence of aliasing, discontinuities, and holes might seem at first to be problematic. These questions will be answered shortly, once the prerequisite material has been developed. Suffice it to say for now that this is not a problem, and no anti-aliasing features of the graphics card are used. In some respects, the discrete nature of the front barrier is a blessing in disguise because it eliminates color shifts as shown in Sub-section 5.2.2. However, the key idea is that the computational

model of the barrier remains continuous, regardless of the physical nature of the hardware use to produce it. Much more on this topic appears in Section 5.3.

5.1.3 Barrier duty cycle adjustment

Duty cycle is the portion of one cycle that is black, expressed as the ratio of black to total cycle. Ordinarily this is .75 or greater. Like the barrier period, the duty cycle is fixed during construction of a static barrier. In Dynallax by contrast, this is adjustable both for the front visible barrier as well as for the virtual barrier. Previously it was known that the duty cycle affects ghosting, and a value of .75 is used to satisfy the Nyquist Sampling Theorem and to leave a guard band between eye channels. Higher values were sometimes used to further reduce ghosting, primarily as a correction for barrier period inaccuracies.

With the ability to vary duty cycle easily in Dynallax, the effects of changing duty cycle can be observed. Figure 18 illustrates this phenomenon. From left to right in Figure 18, the duty cycle is .7, .8, and .9 respectively. Notice how the separation between the centers of the green and blue channels is reduced as the duty cycle is increased. Notice also how the black and cyan guard bands (the regions between pure green and pure blue) change size and shape with various duty cycles.

It should be noted that in Figure 18 and in later tests as well, a color calibration test is used. A single color polygon is rendered for each eye channel, in this case green for the left eye and blue for the right. When the eyes are physically located at the same points in 3D viewer space as the centers of projection (COP) for the perspective views and when the system is well-calibrated, each eye will see a full screen of that respective color. To evaluate the resulting images more objectively and to be able to view both channels simultaneously, a photograph of the screen is captured from a point further back than the center of projection, in this case approximately 2.5 m beyond the COP.

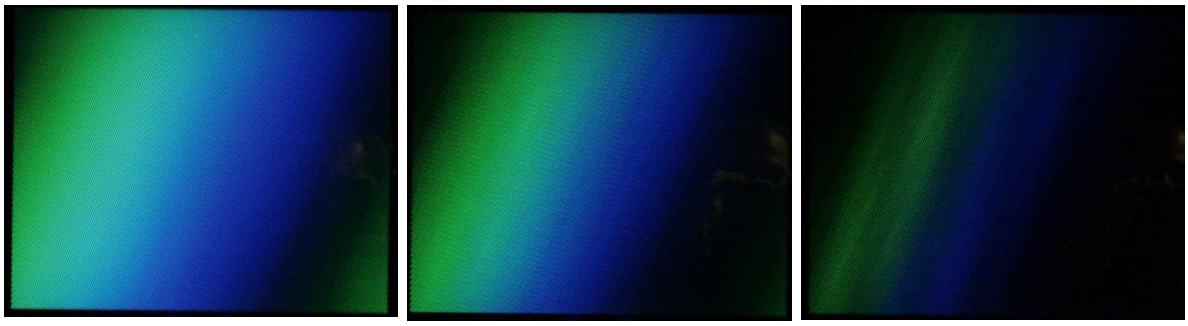


Figure 18. Duty cycles of .7, .8, .9 from left to right

To explain what is occurring in Figure 18, refer to Figure 19. When duty cycle is reduced, the size of the transparent slit increases. Since the parallax barrier slit functions optically as a pinhole lens, the transparent slit size should be maintained as narrow as possible. When it is enlarged, channels become un-focused and guard bands encroach on channels asymmetrically. However, as Figure 18 demonstrates, minimizing the size of the channel slit must be balanced with reducing brightness. In practice, the optimal duty cycle is between .75 and .8.

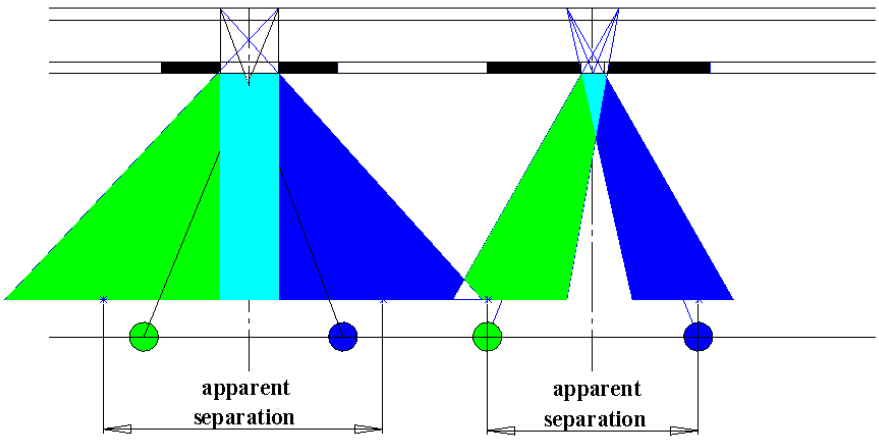


Figure 19. The effect of duty cycle on apparent channel separation

5.1.4 Barrier angle adjustment

The interference between the front display barrier grid and the rear display pixel grid results in a visible Moiré pattern. The size and shape of the Moiré is a function of the barrier period, duty cycle, and tilt orientation

angle of the barrier lines. For a given barrier period and duty cycle, the Moiré pattern can be changed from a coarse pattern to a fine diamond-shaped pattern by rotating the barrier lines through some angle (Winnek, 1968) (van Berkel, 1999). In the past, for a static barrier, finding this optimal angle was time consuming and was accomplished by mechanically mounting a sample barrier in a jig where the barrier could be rotated at any angle in front of the display. Once an optimal angle was found, the actual barrier was mounted to the display at this same angle. The tolerance for this mounting is +/- .25 degrees because the changes in Moiré pattern are quite severe over small angular increments.

With Dynallax, the angle of the physical barrier is changed easily to within any desired angular tolerance, so that finding an angle with relatively small Moiré interference is accomplished in a matter of minutes. Moreover, if various barrier patterns (eg. different period or duty cycle) require their own optimal angular orientation, this is accomplished just as easily. Figure 20 shows three images of Moiré interference patterns using the color calibration test pattern. The left pattern is considered acceptable and is the current single-viewer mode, using a tilt angle of 23.6 degrees. A Moiré pattern is visible, but it contains soft edges and is near-optimal for this configuration of barrier period and duty cycle. The center pattern is a result of a .5 degree change, and dark diamond-shaped pattern becomes visible. The right image shows vertical barrier lines (tilt angle of 0 degrees) and shows extreme Moiré and demonstrates the motivation for rotating the barrier lines.

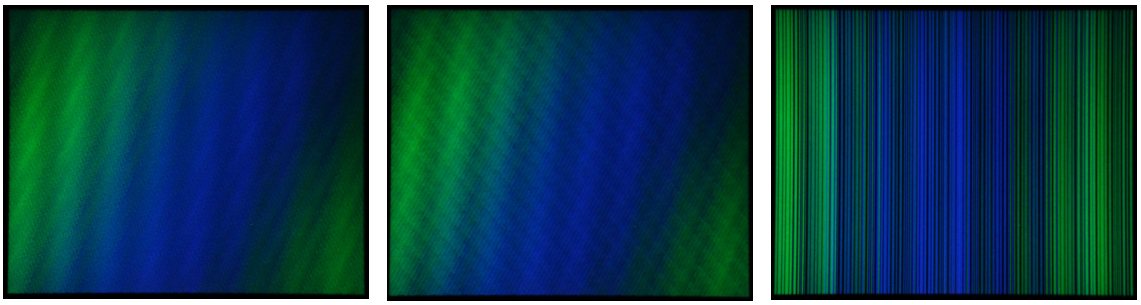


Figure 20. The effect of barrier line orientation on Moiré screen noise

5.2 Side effects

Some convenient and useful side effects result when the front barrier is a dynamic, sub-pixel, discrete barrier as in Dynallax, and two such advantages are explained in this section. The first, in 5.2.1, is the simplification of the process of registering the physical barrier with the virtual barrier computational model, now that the physical barrier is rendered by a computational model. Sub-section 5.2.2 points out another feature, the automatic elimination of color shifts that are ordinarily present in static barrier AS.

5.2.1 Physical and virtual barrier registration

One difficulty in constructing a static barrier system is the registration of the virtual and physical barrier, especially the determination of an accurate barrier period. In Varrier, the virtual barrier period is a function of window size and pixel pitch, and a virtual barrier period needs to be established taking these factors into account. This barrier period is found empirically, and the overall operation of the system is very sensitive to the accuracy of this value. In fact, presently this value is determined to within .003%. At this level of accuracy, ghosting in Varrier is approximately 4% and it has been shown that the ghost level is directly related to the accuracy of the barrier period.

In Dynallax, there is no need for registering the physical and virtual barrier periods because both barriers are produced the same way, through software. Provided that the front and rear displays are constructed from identical LC layers, the barrier period registration problem disappears because the time-consuming step of finding the virtual barrier period is eliminated.

5.2.2 Reduced color shifts

Color shifts in a static barrier AS system are a common occurrence, as in the Varrier system. Before GPU-based modulation algorithms, the solution was a computationally expensive one. The algorithm was required to

execute three passes per eye, with a sub-pixel shift between each pass. This increased rendering time and reduced frame rate threefold. The user or system designer was forced to make a choice between fast performance with poor quality and slow performance with good quality. Dynallax produces little visible color shifting without multiple passes. Pictured in Figure 21, a cross-bar pattern is rendered in Dynallax on the left and Varrier on the right. Both are executing single-pass mode, and the pattern should be pure gray-scale. The Varrier pattern contains color shifts. The cross-bar pattern is another calibration pattern similar to the color calibration pattern, except that now orthogonal lines are used for the eye channels, rather than colors. Again these photographs are taken from beyond the COP, so that both channels can be viewed simultaneously.

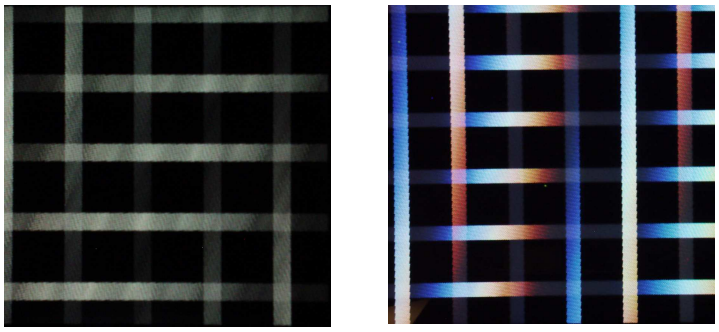


Figure 21. Lack of color shifting in Dynallax (left) compared to Varrier (right)

To explain the elimination of color shifts, consider Figure 22, which illustrates the difference between the ways that light is transmitted in the single pass Varrier, 3-pass Varrier, and Dynallax. In this discussion, it is convenient to ignore the fact that LCD pixels actually transmit light originating from a backlight; assume for the moment that the rear screen pixels are light sources in themselves. No loss of generality results. In each of the three cases, three different light sources generate the light of a single rear screen pixel: a red, green, and blue source. In the 1-pass Varrier, these light sources diverge as they pass through a slit that is 3 sources wide. In the 3-pass Varrier, three light sources send light rays through a slit that is still 3 sources in wide, but it is shifted three times so that its center aligns with the source, thereby maintaining relative parallelism of light rays. Finally, in Dynallax, three light sources send parallel light rays through three different slits, the front screen sub-pixels. In the first case, the divergent light rays result in color separation, while in the latter two cases the parallel light rays result in color cohesion.

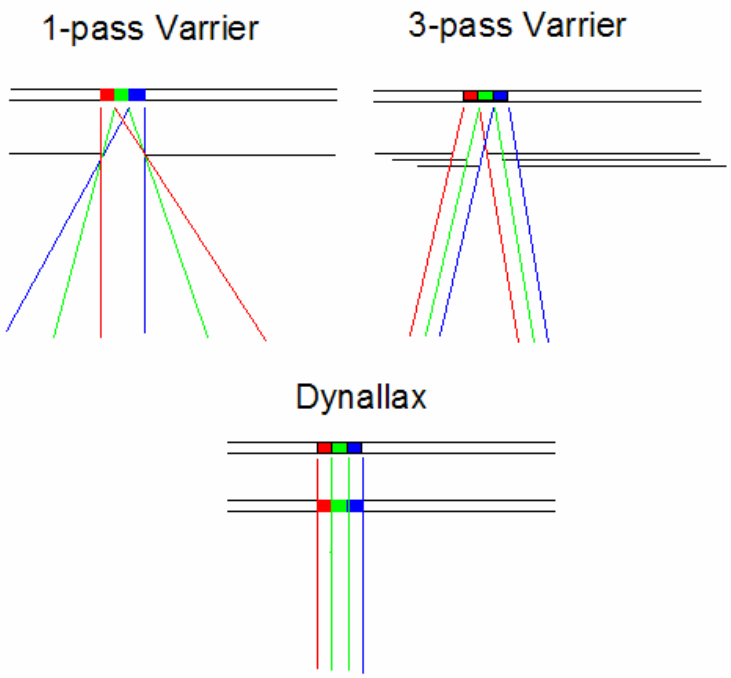


Figure 22. The exacerbation and mitigation of color shifts by various algorithms

5.3 Sub-pixel barrier

Dynallax inherently utilizes a sub-pixel barrier, both in terms of the physical barrier in the front screen and the computational barrier used to interleave images on the rear screen. Static barrier displays such as Varrier contain a physical front barrier, so there is no notion of sub-pixels or pixels in the front barrier of such a display. In a static barrier display, the rear screen interleaving may occur in sub-pixel resolution, depending on the algorithm used. For example, in Varrier the 3/3 and the Varrier Combiner algorithms (Kooima et al., 2007) operate in sub-pixel precision, while the 1/1 algorithm does not.

In general, sub-pixel resolution is desired in the rear screen interleaving algorithm because of increased spatial resolution and reduced color shifts, and the sub-pixel nature of Dynallax's front barrier allows these advantages to be gained with no or little performance penalty. Aside from the physical nature of the front barrier hardware, the underlying sub-pixel virtual barrier computational model can be either discrete or continuous. Sub-section 5.3.1 demonstrates with three examples that a discrete, image-based barrier model is inappropriate for

Dynallax and results in extreme aliasing artifacts of the final image. This motivates the definition of the correct, continuous model in 5.3.2, and the final image interleaving algorithm based on this model is developed in 5.3.3.

5.3.1 Discrete barrier model

Several published works cite the use of a step-wise front barrier composed of discrete red, green, and blue "windows" or sub-pixels (Mashitani et al., 2004), (Schmidt and Grasnick, 2002). In most cases, the barrier is still static, but it is a natural extension to attempt to replicate the same concept with the front LCD screen. The pattern shown in Figure 23 can easily be rendered on the front screen using openGL, either as an image texture or as a procedural texture using the openGL shading language GLSL. Three examples follow where such a discrete texture serves as the barrier model. In all three cases, severe aliasing results despite different attempts to eliminate or reduce the effect. The final conclusion is that such a discrete barrier model is not appropriate for Dynallax.

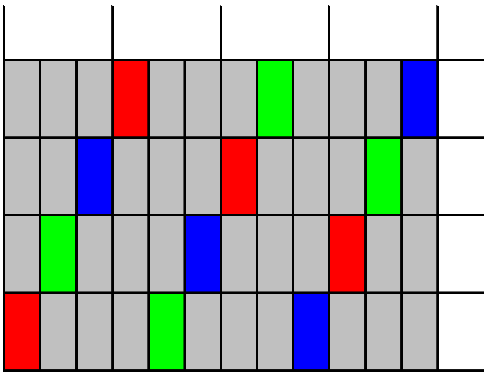


Figure 23. Step-wise discrete barrier

Example 1

A GLSL fragment shader produces the image shown in Figure 23 on the front screen while a second fragment shader modulates image channels on the rear screen based on the visible sub-pixels of the front screen as follows:

```
For each eye {
    For each pixel in rear image {
        Cast a ray from the center of that pixel to the eye and compute the corresponding front pixel location
        Multiply the rear pixel image color by the front pixel color
    }
}
composite the two eye images with an add operation
```

Algorithmic correctness is demonstrated by the following argument: Each front pixel contains exactly one illuminated sub-pixel, so multiplying a rear pixel by a front pixel illuminates exactly one sub-pixel of the rear pixel as well. The final addition operation selects one of the two images since illuminated sub-pixels of the two eye channels form disjoint sets, provided that the viewer is within the usable range of the system. The addition of the channels does not disturb either channel; it simply composites both into one final rendered image.

Although theoretically correct, a severe blocky Moire interference pattern is visible, consisting of approximately 2 inch square blocks. Figure 24 shows a test of just one green channel modulated by this algorithm, and the discontinuities are apparent. When the system is viewed from a normal viewing distance, the discontinuities are overpowering to the extent that they become the single most distinguishing feature of the image.

The pattern is a classic Moire interference of two discrete grids, the front and rear displays, exhibited in square blocks because of identical interference in vertical and horizontal directions. The pixel pattern is square and therefore so is the magnified interference pattern. The problem can be explained in one dimension and then the extension to two dimensions is trivial. Looking at one horizontal scan line of pixels, a discontinuity occurs wherever a front pixel needs to skip over one additional pixel to map to the rear pixel, as shown in Figure 25.

The situation is unavoidable because a perspective projection causes a slightly larger period to occur in the rear screen compared to the front.

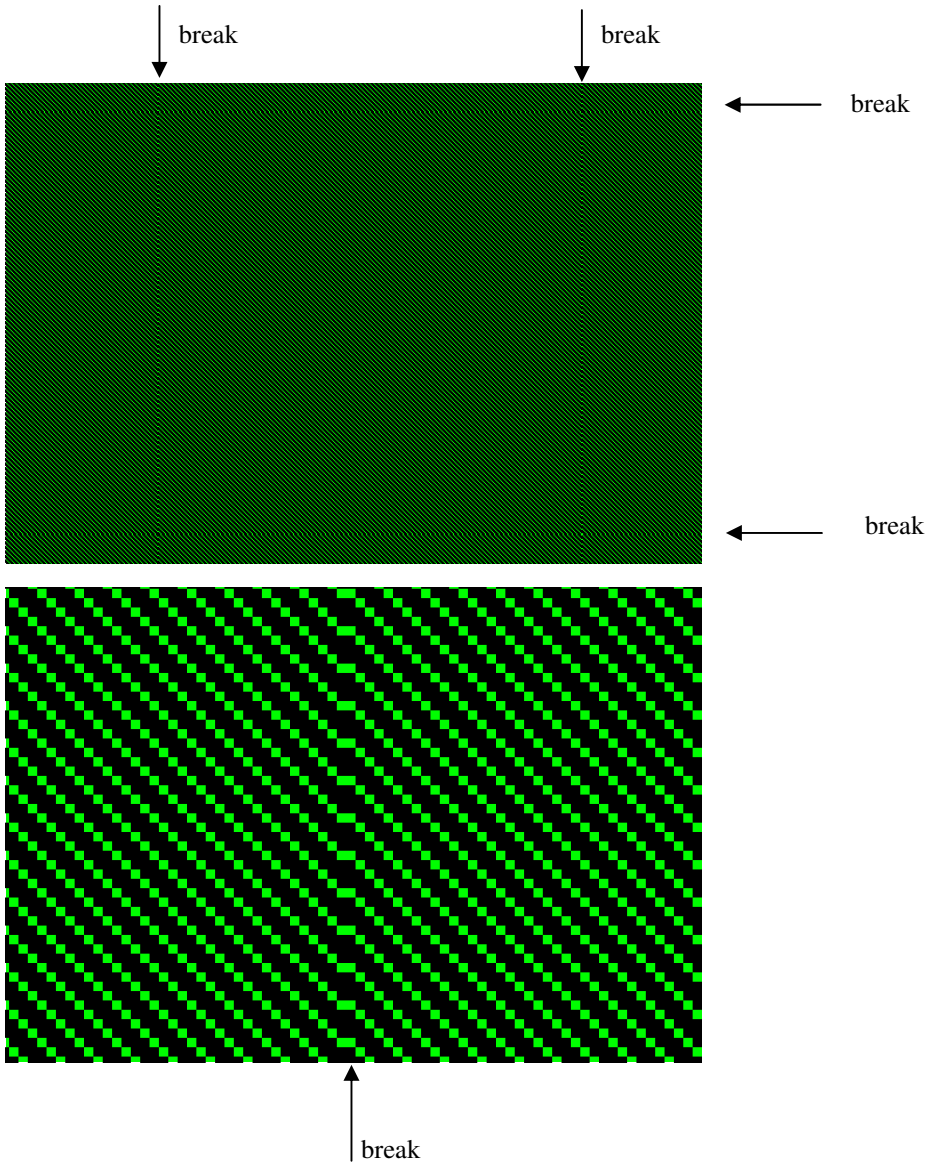


Figure 24. Discontinuities magnified (below) from the image above

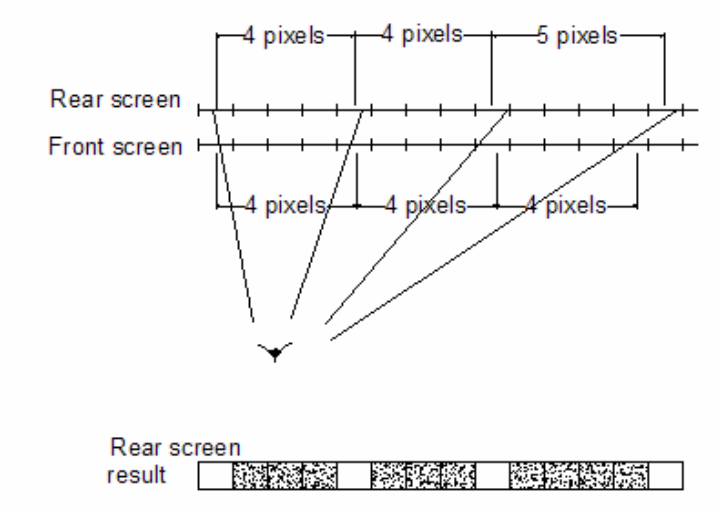


Figure 25. Discrete computation in projecting one grid onto another

Example 2

The all-or-nothing binary algorithm in Example 1 can be modified to sample a 1-pixel size area of the front screen and to mask the rear screen according to the colors covered by this area.

The rear pixel is still multiplied by the resulting mask, but now the mask is computed as a weighted average of the colors covered in the front screen, weighted by their percentage of coverage. The left and right eye images still computed separately and then added. The result is that the sharp discontinuities have been replaced by softened brighter and darker regions, as shown in Figure 26, but they have not disappeared.

Figure 26. Smoothing of artifacts by area averaging

Example 3

Rather than smoothing as in Example 2, the texture is projected from the front plane to the back via an openGL texture projection and the result is sampled by openGL. Then, the scene projection is masked according to the projected texture with a similar multiply operation as before. Figure 27 shows screen shots of the resulting projected barrier rendered on the rear screen, although the problem is more severe when viewed by the eye. The sharp square pattern that results as pixels skip over one in the hard pattern in the left image can be softened in the right image by filtering using openGL linear interpolation.

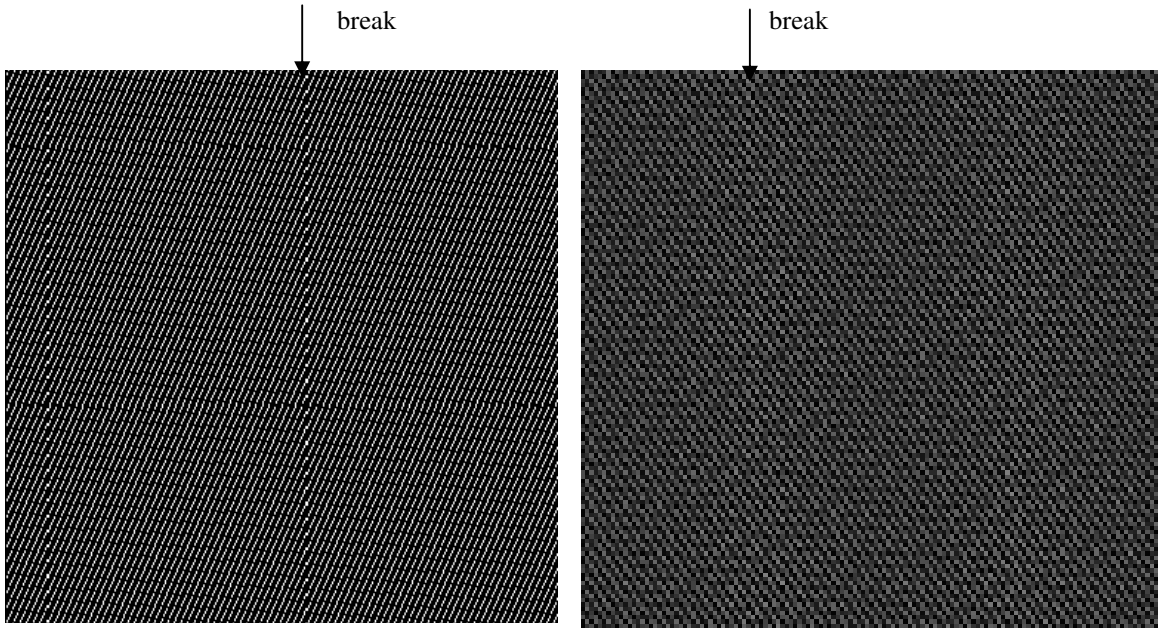


Figure 27. Results of sampling using openGL texture projection with (right) and without (left) filtering

Conclusion

The problem is not in the algorithmic details, but in a common underlying fact: an image-based entity that has already been discretized is being re-discretized during the projection from the front screen to the back. Each of the three examples accomplished this re-discretization a different way (binary, weighted average, and openGL

texture projection) with a similar result. Aliasing caused by the projection of one discrete image onto a second discrete grid can be mitigated by filtering, but not eliminated.

The solution is to avoid re-discretization altogether. The barrier computational model must remain continuous and be rasterized only once, by the graphics hardware at the end of the rendering pipeline. This is why the static Varrier works correctly. There, a continuous floating point polygonal model of a barrier is maintained and then projected onto the display only once. This is a valuable finding and bears repeating: image based discrete barriers are incompatible with Varrier and Dynallax.

5.3.2 Continuous barrier model

The model of the barrier is continuous in floating point space in order to maintain the single-discretization principle described above. A barrier is defined by the following continuous parameters:

a. Barrier period or line spacing in lines per foot (lpf)
b. Angular orientation (line tilt) in degrees from vertical
c. Duty cycle, or fraction of opaque to total period
d. Lateral shift (left-right) along the plane of the display screen from some reference point such as the screen center
e. Optical distance that the barrier is located in front of the rear display screen

The last two parameters can be considered as the position of the barrier in 3D space, and thus the barrier is defined by its position, orientation, period, and duty cycle, all in continuous floating point world space. This is similar to the Varrier 1/1 and 3/3 barrier models, with the following key differences:

a. In Dynallax the same model is maintained for the physical front barrier and for the virtual barrier used to modulate the scene of the rear display, whereas in Varrier the front barrier is physical and has no computation associated with it.

b. The barrier model in Varrier 1/1 and 3/3 is maintained as a collection of polygons within the virtual world because of the depth buffer algorithm used to modulate the scene. In Dynallax, no such polygonal set exists because modulation occurs using a fragment shader algorithm (see algorithm below). The computational model *is* the set of five parameters above.

c. The Dynallax computational model permits (and encourages) the dynamic modification of the barrier model as it is being used in order to maintain optimal performance and expand the capabilities of the system, as compared to Varrier.

5.3.3 Final algorithm

Based on the above model, the final algorithm is executed partly in openGL and partly in a GLSL (Rost, 2004) fragment shader. One shader program handles the case of two eye channels (one viewer) and a separate shader program handles the case of four eye channels (two viewers). The algorithm for either case can be generalized as follows:

```
For each eye {                  // performed in opengl application
        render the perspective into a separate offscreen texture buffer
}
For each pixel {                // performed in fragment shader
        Mask each eye view by a perspective projected barrier on a sub-pixel component basis
        Take component-wise maximum of the masked eye channels and assign the pixel to this set of 3
        components
}
```

The perspective projection of the barrier from the front screen to the rear can be computed with a simple scaling of the period and a translation of the entire pattern. In the general case, perspective projection is a non-linear function. However, in the special case that all projected points originate at the same distance from the display, the function is conveniently linear. This is the case in a parallax barrier. The scaling and translation are computed as follows in Figure 28:

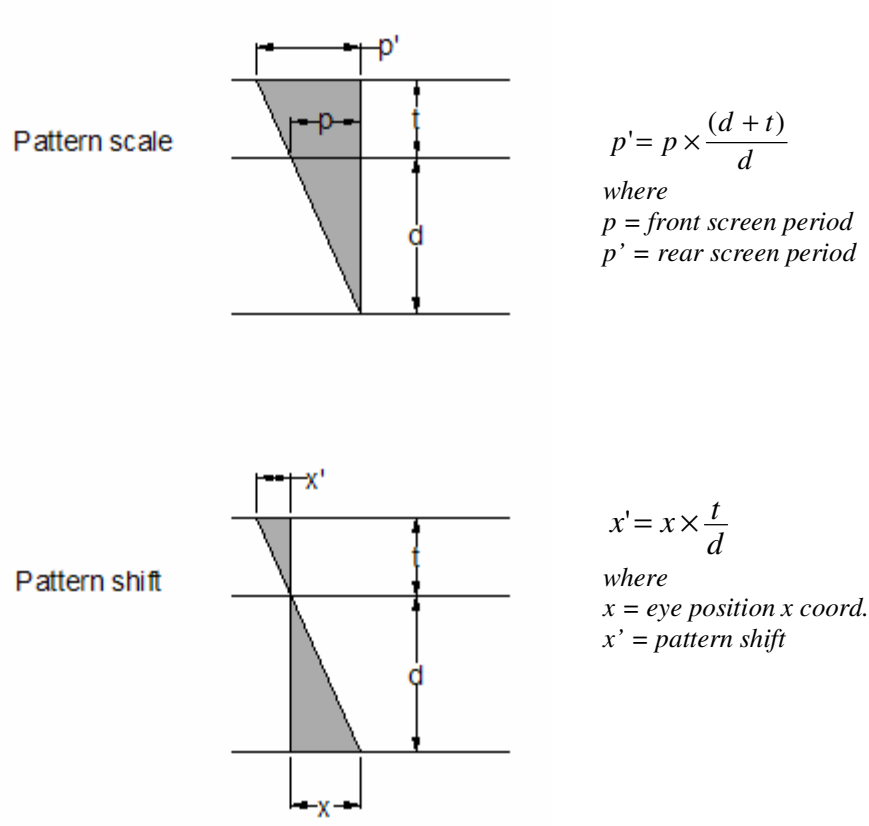


Figure 28. Derivation of scaling and translation due to perspective projection of front barrier to rear

Figure 29 continues the derivation of the barrier fragment geometry. The lower half of Figure 29 shows a given horizontal scan line within the space of the screen, and the line tilt angle θ. An offset of the pattern at the y coordinate of the scan line is computed by simple trigonometry as $y * \tan \theta$. The horizontal period of the pattern is computed by dividing the period p' of Figure 28 by cos θ, as shown in the upper half of Figure 29. Then for

any fragment coordinate within that scan line, the fraction of the *(fragment x coord + offset) / (p' / cos θ)* is computed. This maps the fragment x position to the range of [0,1], and then the step function shown in the top of Figure 29 is applied with the edge of the step at the duty cycle (eg. .75) to decided whether the fragment is white or black (barrier is transparent or opaque).

The above idea is then expanded to address sub-pixels in both the drawing of the front barrier and the modulation of the rear barrier. The perspective scale and translate functions remain the same, but the test whether to draw a pixel in the front screen or to modulate a pixel in the rear screen is replaced by 3 tests for the red component, green component, blue component.

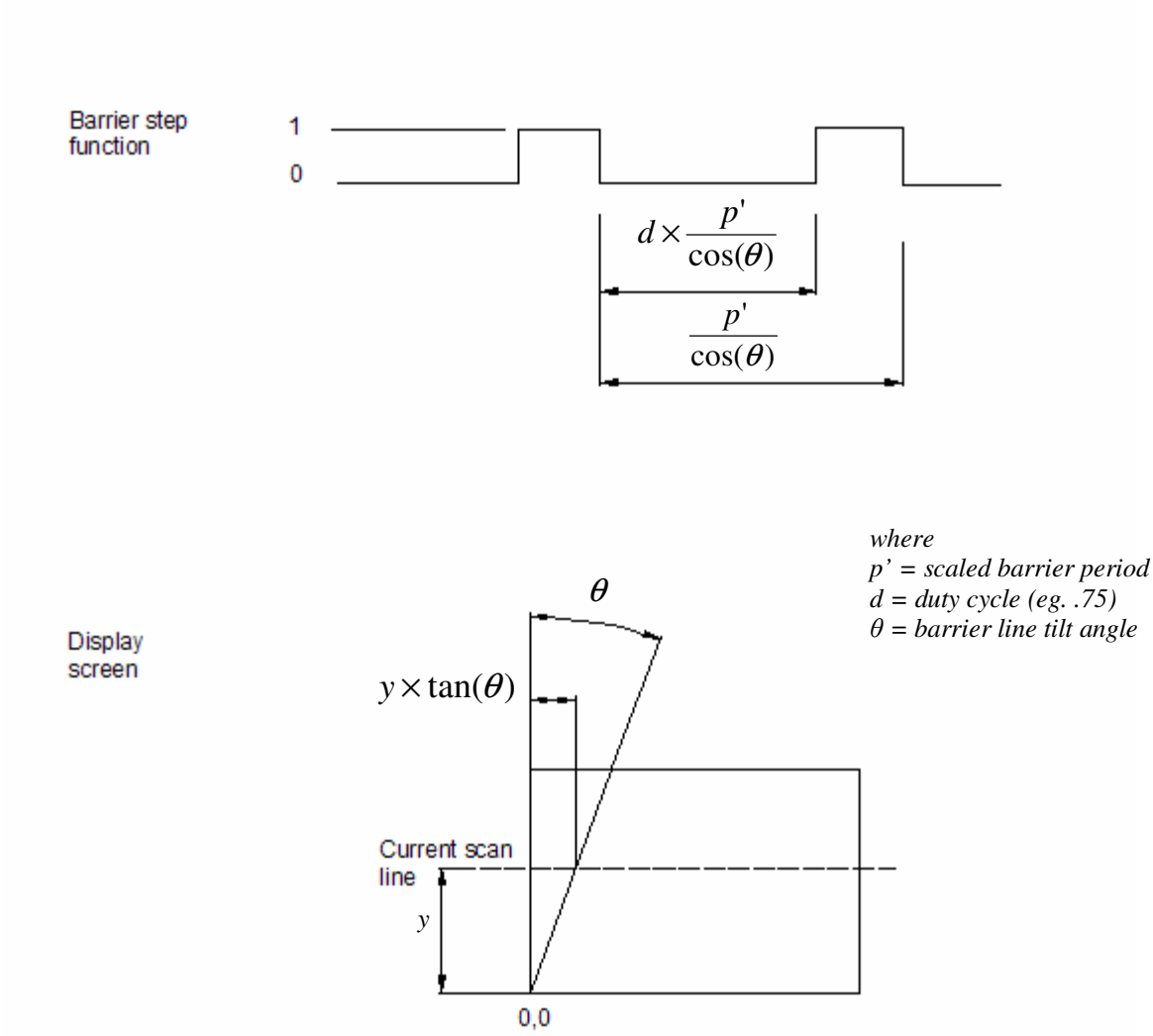


Figure 29. Derivation of barrier step function

Next, the concepts from Figure 29 are used to compute constants *k1, k2,* and *k3*, which are the interface to the fragment shader programs. In the following equations, screen space is defined as the right handed coordinate system whose origin is in the center of the front barrier screen, whose x direction is along the horizontal direction of the screen, whose y direction is along the vertical direction of screen, and whose z direction is normal to the plane of the screen. The units of screen space are pixels. Given the following system parameters,

d = duty cycle

p = barrier period in world units

shift = barrier shift in world units

θ = line tilt angle

opt = optical thickness in world units

width = screen width in pixels

w = screen width in real world units

norm_dist = normal distance from eyes to front barrier plane in world units

eye_x, eye_y = x,y location of eyes in screen space

Let us define the following inputs to the fragment shader programs:

k1 = tangent of line tilt

k2 = shift in the x direction of screen space

k3 = horizontal period p in the x direction of screen space

The inputs *k1, k2,* and *k3* are computed as follows for the front and rear barriers. For the rear barrier, *k2* and *k3* are actually a vector, one value for each of the user(s) eyes.

Front barrier parameters:

$$k1 = \tan\theta$$
$$k2 = shift \times \frac{width}{w} \qquad (2)$$
$$k3 = \frac{p}{\cos\theta} \times \frac{width}{w}$$

Rear barrier parameters:

$$k1 = \tan\theta$$
$$k2 = \left(\frac{norm\_dist + opt}{norm\_dist} - 1\right) \times \left(eye_x - eye_y \times k1\right) + \left(\frac{norm\_dist + opt}{norm\_dist}\right) \times shift \times \frac{width}{w} \qquad (3)$$
$$k3 = \left(\frac{norm\_dist + opt}{norm\_dist}\right) \times \frac{p}{\cos\theta} \times \frac{width}{w}$$

Results

The front barrier is shown in Figure 30. The channel width is approximately 2 sub-pixels, although the screen capture shows only pixel resolution, hence the color combinations. For example, a cyan pixel actually represents a channel covering the green and blue sub-pixels. Meanwhile, Figure 31 is a screen capture of the rear screen during cross-bar calibration mode. This screen shot shows sub-pixel modulation of white cross bars, vertical for the left eye and horizontal for the right. Where the bars cross, both eye patterns are visible side-by side. A 1-pixel guard band exists between pairs of channels, and a smaller guard band exists between channels, one or two sub-pixels, although the pixel-resolution screen shot does not capture such fine resolution.

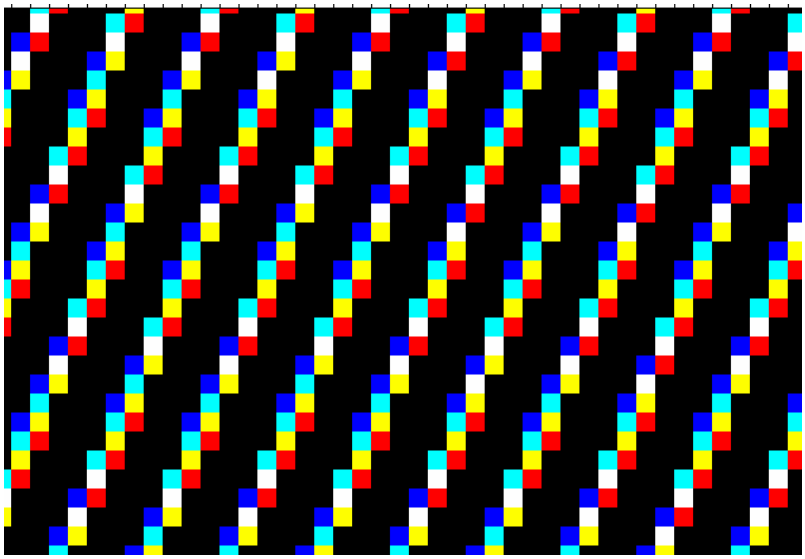


Figure 30. Sub-pixel channels illustrated in front screen capture

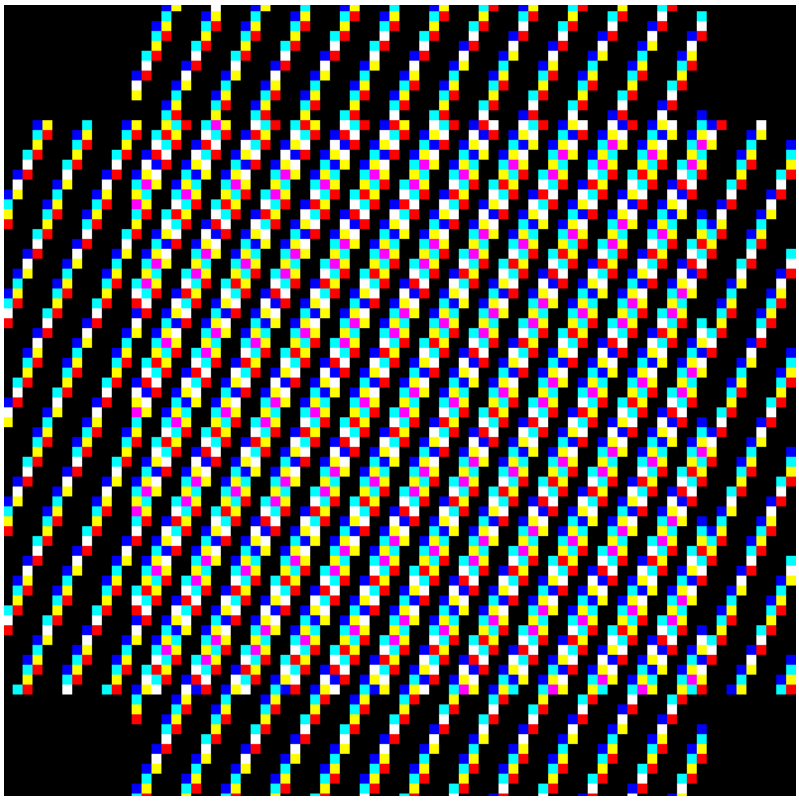


Figure 31. Cross bar pattern in sub-pixel scale captured on the rear screen

5.4 View distance and channel separation

Two problems are addressed here. One is a limitation of static parallax barrier systems in general, namely that the near viewing distance is limited to some pre-determined value. Often users desire to be closer than this limit, in an attempt to view some detail of the VR scene more closely or to have the VR scene encompass more of their field of view. The second issue is particular to Dynallax, namely that the usable view distance range of Dynallax is limited compared to other systems such as Varrier. View distance is the z distance in the direction normal to the display plane. Fortunately, both challenges are tractable and have been solved in the final Dynallax implemenation with an elegant, simple method. Some background information on the view distance – channel separation problem is presented in Section 5.4.1. The problem is defined more rigorously with equations in Section 5.4.2, and the optimal view distance equation leads directly to the algorithm in Section 5.4.3. View distance results are presented in Section 5.4.4.

5.4.1 The effect of view distance on channel output

The scene that is rendered on the rear screen is composed of interleaved strips of eye channels. Assume for this discussion that one viewer, two eye channels are employed and the left eye scene consists of only a green background and the right eye scene consists of only a blue background, the usual green-blue color test pattern. No loss of generality is imposed on the number of channels or their content by these assumptions; they only frame the discussion in practical terms. Within each cycle of green-blue content rendered on the rear screen, some additional black region(s) also exist. These black regions are called guard bands, and serve to delineate the two eye channels and help prevent cross-talk or ghosting. In fact, the Nyquist sampling theorem requires their existence and further requires that the minimum amount of guard band is equal to the amount of green and blue.

The relative location of black guard band to blue-green content changes with view distance for a given barrier period. See Figure 32, which depicts the rear screen rendering at different z distances of the viewer. When the viewer is at the minimum usable view distance of the system (the near distance), the entire guard band is situated

between green and blue channels and no guard band exists between adjacent cycles. This is the case in the left image. When the viewer is at the optimal view distance (the sweet spot), guard bands are equally distributed between the channels of one cycle and adjacent cycles. This is the middle image. Finally, when the viewer is at the maximum view distance (the far distance), all of the guard band exists between cycles and none exists between channels of the same cycle, as in the right image.

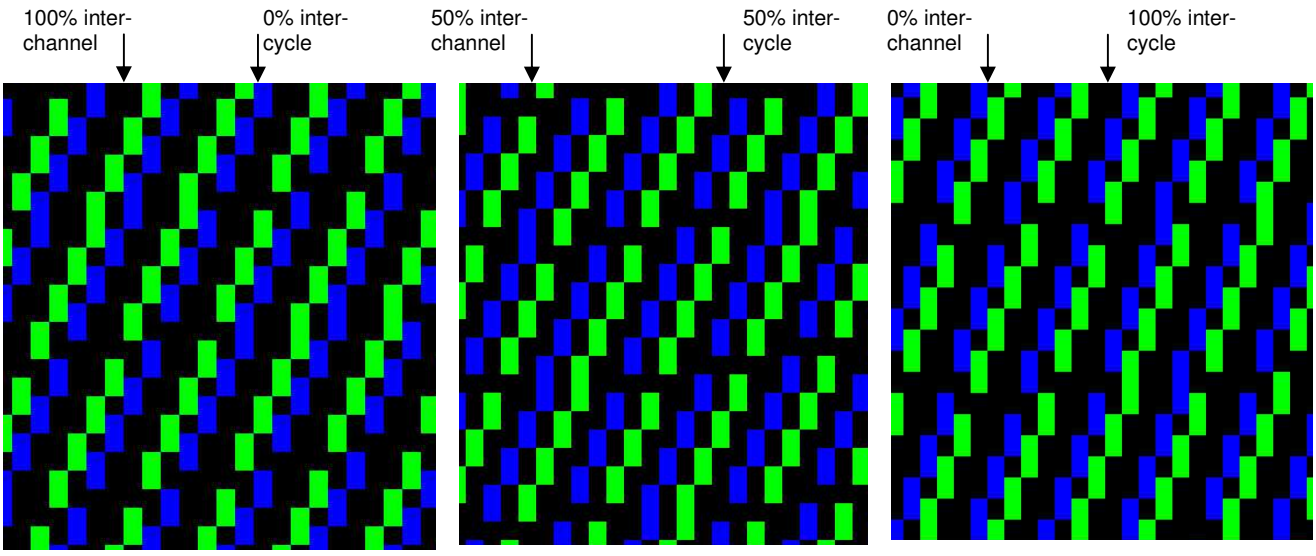


Figure 32. Distribution of channels and guard bands at near (left), optimal (center), and far (right) distances in static barrier display

Figure 33 illustrates the same re-distribution of channels and guard bands, but this time at the position of the viewer. At the near distance, the entire guard band exists between the viewer's eyes and none of the guard band is between adjacent lobes. At the sweet spot, guard band is equally distributed between the eyes and adjacent lobes, while at the far distance, none of the guard band exists between the eyes and all exists between adjacent lobes. Clearly, in a static system, once the viewer moves closer to the display than the near distance, ghosting results as channels begin to overlap. The amount of screen "real estate" for channels to occupy is a direct function of the barrier period. Larger barrier periods permit more space for channels to approach each other, and hence permit the user to move closer to the display.

In traditional Varrier, the usable Z view distance range is at least 2-3 feet, usually limited by tracking rather than the Varrier optics. However, the observed Z working range of Dynallax is much tighter, approximately 1 foot. When moving beyond this limited range, Dynallax is observed to produce channels with incorrect separation; channels are no longer separated by the interocular distance from which they were computed. This occurs sooner than predicted by equations 3 and 4 below, in both the near and far limits. In fact, the channel separation linearly increases from approximately 25% too narrow to 25% too wide as the user moves from 1 foot in front of the sweet spot to 1 foot beyond it. At the same time, the pattern of guard bands does not appear distinctly distributed. Rather than guard bands "sliding" from between the lobes to between channels as shown in Figures 32 and 33, they encroach upon the green-blue channels and affect channel separation and clarity.

When screen shots of the rear screen are taken during these conditions and are compared to static barrier conditions, one confirms that Dynallax is computationally correct. It is hypothesized that the colored sub-pixel nature of the front barrier is incorrectly steering channels, with an error that is linear in the distance from the sweet spot. As will be shown in Section 5.4.3 however, the cause is inconsequential because the Dynallax algorithm is designed to always maintain optimality so that every position is the sweet spot.

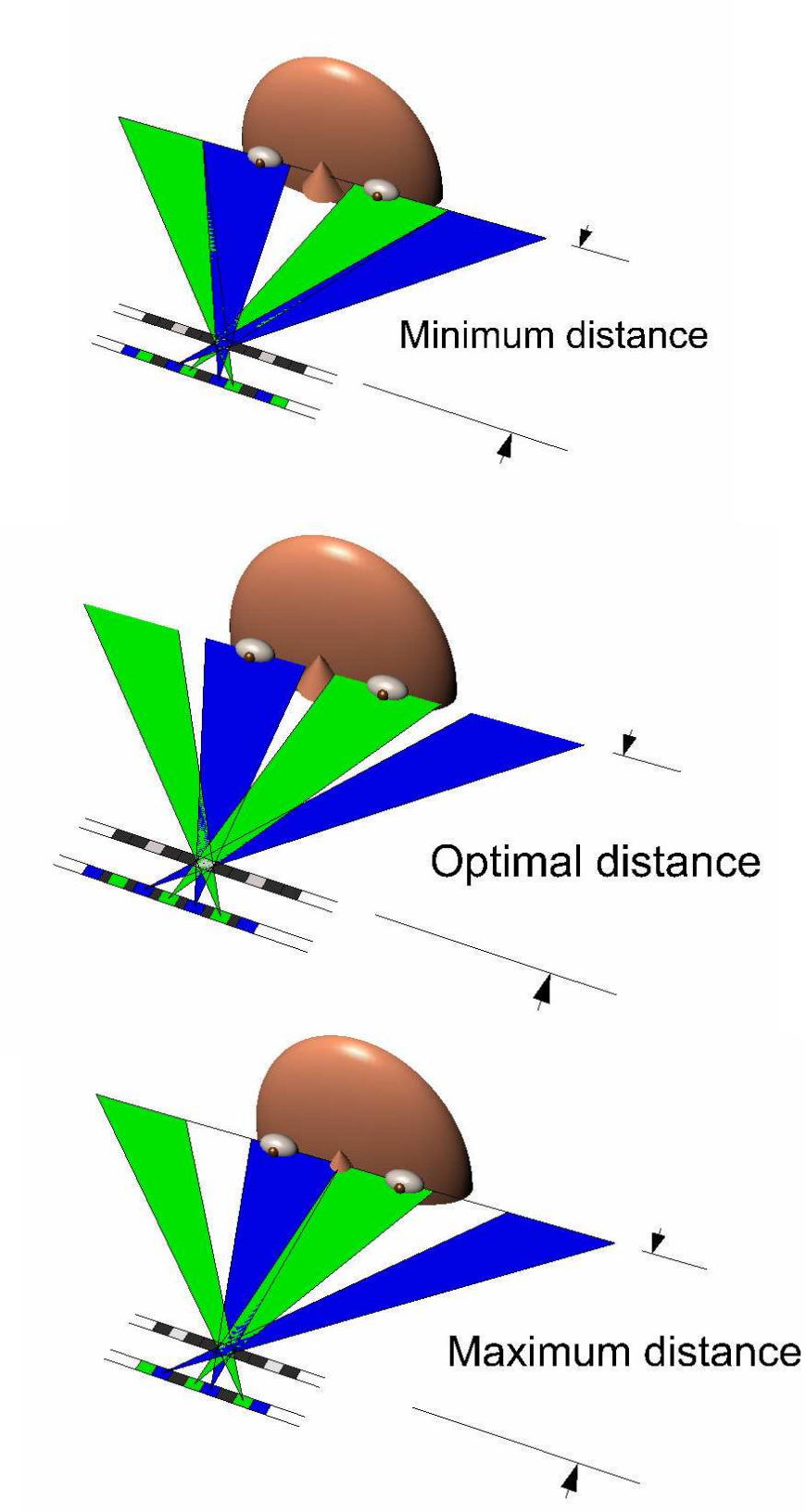


Figure 33. Distribution of guard bands and channels at the viewer location for a fixed barrier system

5.4.2 View distance equations

Sandin et al. (Sandin et al., 2005) derived the following view distance equations for a static barrier display with pixel-scale channel resolution:

$$z_{\max} = \frac{t \cdot e}{s} \qquad (3)$$
$$z_{\min} = t \cdot \left(\frac{e - p}{p - s} \right)$$

Where e is the interocular distance, p is the barrier period, s is the pixel pitch, and t is the optical thickness. Equation 3 proves that Dynallax has no minimum view distance because the minimum view distance is a function of p, and p is variable in a dynamic barrier system.

The equations in (3) can be converted to the sub-pixel scale barrier by replacing the pixel width s by the channel width. The channel width, or width of the transparent portion of a cycle, is generally ¼ of the total cycle, and substituting $p/4$ for s yields the following:

$$z_{\max} = \frac{4t \cdot e}{p} \qquad (4)$$
$$z_{\min} = 4t \cdot \left(\frac{e - p}{p - s} \right)$$

In Sandin et al., the optimal view distance was computed as the mean of the two extrema. However, other criteria exist to define optimality and the optimum view distance may differ, since the equations are non-linear. An optical definition of the optimality is the view distance where channels and guard bands are equally distributed, as in the center image of Figures 32 and 33. Figure 34 formulates the necessary geometry for the optimal view distance:

$$z_{opt} = \frac{2t(e - p)}{p} \qquad (5)$$

Since $2e >> p$, and $d >> t$, p is approximately proportional to $2e$ by the factor d / t, or

$$\frac{2e}{p} \approx \frac{d}{t} \qquad (6)$$

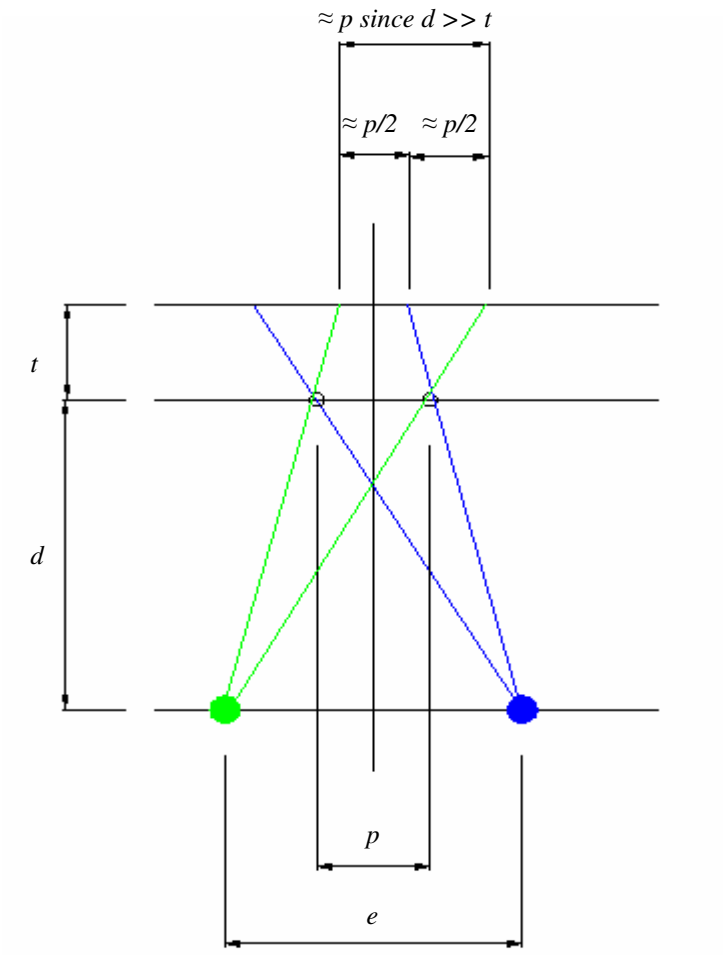


Figure 34. Optimal view distance derivation

5.4.3 Optimal view distance algorithm

Given the narrow range of view distance in Dynallax, the desire to eliminate the minimum view distance limitation of static barrier systems, and the optimal view distance equation 5, the solution is a natural one. The

barrier period is always maintained to produce an optimal condition using a real-time controller algorithm. In practice, the barrier period is computed at every frame, but only changed when it differs from the previous period by some threshold amount, eg. 10%. This reduces the amount of visual flicker seen by the viewer during period changes. When 2 viewers are present, the view distance of both viewers is averaged.

The closer the viewer approaches the display, the larger the barrier period becomes. In theory, there is no near limit on view distance, however at very near distances, the barrier period becomes large enough to be noticeably distracting. Two solutions exist. First, the physical separation between front and rear panels can be reduced. The separation is a static physical parameter of the design and construction of the system rather than an adjustment to be performed during use. Reducing the physical space between screens also reduces the far viewing distance. The other approach is to clamp the dynamic range of the barrier period to some maximum value, and when the user exceeds this value, the channel quality begins to slowly deteriorate as explained above. In practice, the dynamic barrier period controller increases the working range of Dynallax from 1 foot (without dynamic control) to 3 feet (with dynamic control), a 300% increase without the barrier period becoming noticeably large.

When the controller module instructs the front and rear display nodes to change barrier period, they must do so in synchronization with each other, in order to minimize distraction to the viewer. The barrier rendered on the front display must be identical to that used to compute the rear display image. If synchronization is not enforced, a visible flicker in the display is quite noticeable because the two displays have dissimilar barriers for a fraction of a second. Screen synchronization is enforced with a barrier command that is part of the message passing architecture. The MPICH-2 (MPICH, 2005) message passing scheme employed in Dynallax has such an MPI_Barrier() command. Processes block when reaching this command until all processes have reached it; this rendezvous mechanism is a common tool in distributed architectures used to synchronize processes. This command is placed the end of the drawing function in each slave node.

As the controller modifies the barrier period according to equation 5, it maintains the same barrier duty cycle, eg. .75. Hence, the physical width of the transparent portion of the period changes with the total barrier width.

This is an important point to note; in contrast other modes such as the two-viewer mode will modify both the barrier period and duty cycle in order to maintain the same physical transparent width.

One other barrier parameter that does need to change with the barrier period is barrier shift, although this may not be immediately obvious. Shift is the horizontal displacement of the barrier pattern, in the x direction of screen space, measured in world units. It is generally non-zero in order to adjust for non-exact placement of the two LCD screens with each other, as well as a result of the rapid view steering controller module (see Section 5.5). At first it may seem that barrier period is independent of shift, but the two are actually linearly related. The visual output of the shift is modulo the barrier period; ie, the shift repeats in multiples of the barrier period. The shift (in world units) corresponds to some fraction of a barrier cycle, and when the view distance controller reduces the barrier period by some factor, the shift must be reduced by the same factor:

$$shift_{new} = shift_{old} \cdot \frac{period_{new}}{period_{old}} \qquad (7)$$

While there is no theoretical near limit to the system, there is still a far limit. It occurs when a channel becomes narrower than a front screen pixel or sub-pixel, depending on whether the front barrier is colored, as explained in Section 5.3.4. Assume for this discussion that the front screen is colored, limiting the channel width to one pixel. The barrier period is 4 times the channel width (for .75 duty cycle), or 4 pixels at the far limit. Equation 1 assumes that channels are adjacent at the far limit, but Dynallax's increased sensitivity to optimality requires that a one pixel guard band be maintained, so specifically for Dynallax, the far limit is restricted by a factor of 2:

$$z_{\max} = e \cdot \frac{t}{2s} \qquad (8)$$

where s is the width of a pixel or sub-pixel, depending on the makeup of the front screen. With the current colored front screen, Equation 8 results in a far limit for Dynallax of 34 inches.

5.4.4 Results

The ghost level in the vicinity of the near limit can be measured with a photometer. One eye channel is rendered completely white while the other is rendered completely black, and the difference in light levels read by the photometer is converted to a percentage of crosstalk, or ghost level. In a static barrier system, the ghost level is a minimum at the optimal view distance defined by equation 5 and increases non-linearly as the near limit is approached. This is a limitation of the static barrier system, as viewers often desire to be near the screen to view details of the VR scene content.

Figure 35 depicts the improvement in ghost level at near distances in Dynallax compared to other static barrier implementations such as the 35-panel Cylindrical Varrier (Sandin et al., 2005) and the Personal Varrier (Peterka et al., 2006). The Personal Varrier is a seated display whose static barrier is tuned to closer distances than the Cylindrical Varrier, but the pattern is similar for both static displays. Both curves spike upward at near distances. Dynallax, by contrast, maintains optimal conditions over a range of distances. In fact, Dynallax actually improves at near distances, as seen in Figure 35. The larger barrier period imposed by the controller algorithm steers channels more accurately because more pixels are covered by each channel, reducing aliasing due to sampling errors. Moreover, the shorter view distance leaves less distance for any steering errors that do exist to deviate from the eye positions.

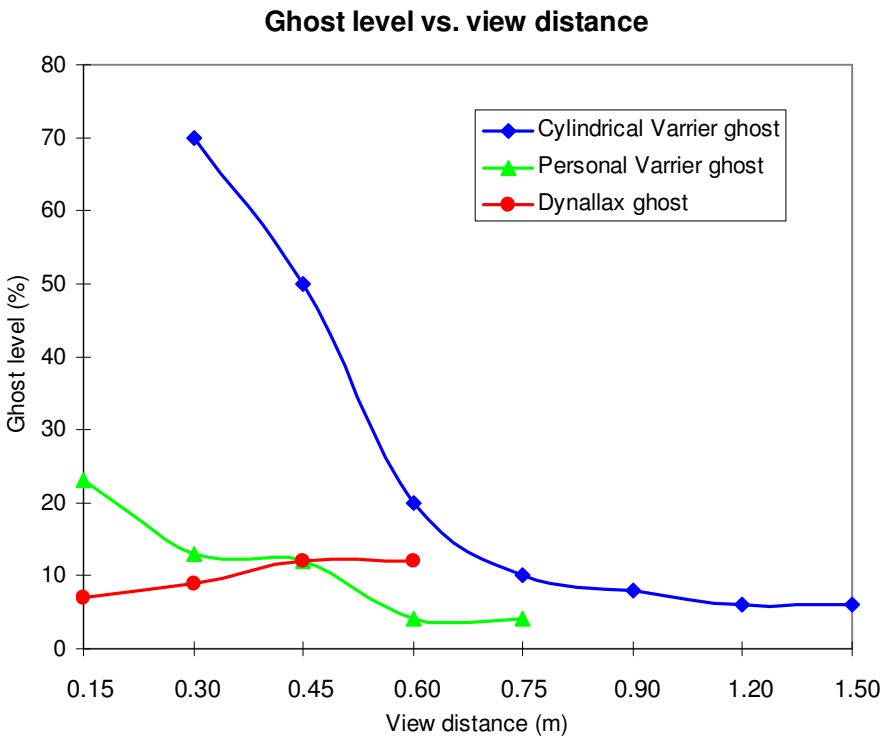


Figure 35. Dynallax ghost level improvement at near view distances compared to static barrier systems

5.5 Rapid channel steering

Fixed barrier AS systems are overly sensitive to rapid, or even moderate-speed head movement. This latency has several components, for example frame rendering time and communication time. In (Sandin et al., 2005), end to end latency has been measured to be on the order of 80 msec. This is the time from which the user moves his or her head until that movement is registered by an updated image on the display. The error that is visible during head movements in static barrier displays is related to the distance a viewer can move the head during this 80 ms interval, before the system can update the display. In everyday contexts, this may seem inconsequential, even negligible. Even in field sequential VR such as the CAVE (Cruz-Neira et al., 1992, 1993), similar latencies have been reported (He et al., 2000). However, latency is more noticeable to the viewer in parallax barrier AS than in other VR methods. Since channels are spatially multiplexed, both channels are visible in different locations within the same image. When head movements out-run the system, the eyes will perceive the incorrect channel or the black guard bands that separate channels. This results in dark flickering, ghosting or even pseudo-scopic

vision. These effects are visually disturbing, and can dominate the VR experience. There is no remedy for this phenomenon in a static barrier system.

Fortunately, the dynamic barrier presents an opportunity to mitigate the sensitivity to AS latency. In general, scientific visualizations and other VR content is data-heavy and it is difficult to maintain interactive frame rates of 60, 30, or sometimes even 15 Hz. Data sets are continuously increasing in size and complexity, and graphics hardware advances are always met with even greater demand for even more capacity. This is the nature of VR, and computing in general. Frame rate is usually sacrificed for complexity and image quality. The good news is that Dynallax contains two addressable image screens, and the aforementioned loading (often over-loading) applies to only the rear screen. The front screen only renders the barrier, and has a constant computational O(1) complexity. The front screen can easily run at a 60 Hz frame rate on modern graphics hardware, regardless of the graphics load on the rear screen.

This unequal load on the two screens permits a rapid steering mode to mitigate the sensitivity to latency. While the rear screen continues its slower rendering process, the faster front screen shifts the barrier to steer channels to the viewer's location at 60 Hz. While this is occurring, the perspective being rendered is not current, but the steering of channels is. As soon as a new perspective is available on the rear screen, it becomes visible to the viewer. In practice, the interlacing of steered old and new perspectives is seamless and the net effect is that fewer artifacts are visible. In rapid steering mode, the front and rear screens are de-coupled and are running in an asynchronous communication mode, both proceeding at their fastest possible rates. The rapid steering control module monitors this behavior and sets the shift of the front screen according to equation 9:

$$s = e_x \cdot \frac{t}{t + d} \tag{9}$$

where:

s = barrier shift amount

e_x = horizontal head movement distance

t = optical thickness between front and rear screens

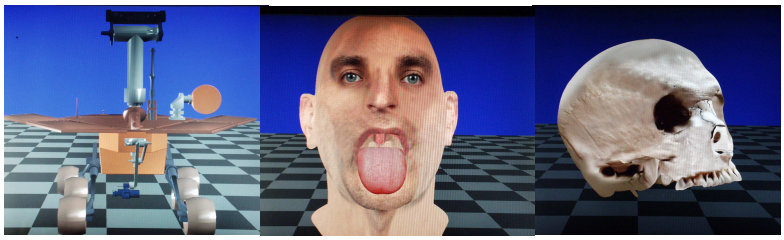
d = normal distance from eyes to front screen plane

Table VI presents frame rate performance data for the front and rear screens for various scene complexities, both with rapid steering mode enabled and disabled. Images of the three datasets in Table VI, "Mars rover," "head," and "skull," appear below the table. With rapid steering disabled, both screens operate in lockstep at the same frame rate. However, with rapid steering enabled, each screen runs at its fastest rate, rendering whatever data is currently available at the time. The front screen pattern shifts left and right according to equation 9.

TABLE VI

FRAME RATES WITH RAPID STEERING ENABLED / DISABLED

Scene model	**single polygon**	**Mars rover**	**head**	**skull**
# vertices	4	15K	130K	220K
front screen frame rate, synchronous	30 Hz	30 Hz	10 Hz	3 Hz
front screen frame rate, asynchronous	50 Hz	50 Hz	50 Hz	50 Hz
rear screen frame rate	30 Hz	30 Hz	10 Hz	3 Hz



To analyze the data in Table VI another way, one may compute the maximum head velocity permissible with and without rapid steering mode. The result is not as dramatic as the frame rate increases in Table VI, since now other communication overhead must be added that is constant in both modes. Assume this constant overhead is

65 ms, as in (Sandin et al, 2005). Further, assume that head velocity is limited by a maximum movement of one half the interocular distance, or 32 mm, during the time that a frame is displayed. This is a realistic criterion, since at that point the eyes have moved completely into black guard band regions before the system has updated the graphics. So, if the time taken to render a frame according to Table VI is added to the communication latency, the resulting permissible head velocity is:

with rapid steering: v = 32 mm / (65 + 20 ms) = .38 m / s

without rapid steering: v = 32 mm / (65 + 333 ms) = .08 m / s

The resulting increase is a factor of greater than 4 times.

5.6 Two viewer mode

Another feature of a dynamic barrier is the ability to support two tracked viewers simultaneously, that is, to spatially multiplex four channels instead of two. This is impractical or impossible with a static barrier for two reasons. The period of the barrier needs to be at least twice as large compared to a single viewer. A static barrier could be designed with a larger period, but if it were sometimes used with only a single viewer, as is the case in Dynallax, then half of the available screen resolution would be wasted. Screen resolution is a valuable commodity in any VR display, but especially in a parallax barrier display. While this first problem is inefficient, the second problem is intractable. When two viewers head positions vary, a static barrier will produce conflicts between channels. Only a dynamic barrier can avoid these conflicts by varying its period in real time.

The controller module that already monitors barrier period due to view distance and barrier shift due to rapid head movements takes on a third function now: to monitor the barrier when two viewers are present. This section explains how this is accomplished, presents the math underlying the algorithm, and analyzes the results of two viewer mode. To motivate the discussion, Figure 36 shows four channels efficiently multiplexed for a given fixed pair of viewer positions. This is an extension of the familiar green-blue color test pattern that has appeared

in numerous figures before. The screen is again photographed from a far distance, much farther than the COPs, so that all channels are visible. However, now four colors are used to represent the stereo channels of two viewers. One viewer is receiving yellow and magenta channels in left and right eye, while the other viewer is receiving green and blue, respectively.

Figure 36. Demonstration of two-viewer mode at fixed viewer positions

Section 5.6 is a rather lengthy and detailed part of the dissertation, and is organized as follows. Sub-section 5.6.1 begins by introducing the main problem to be solved in two-viewer mode: avoiding conflicts between virtual lobes. Next, a general minimization algorithm is presented in 5.6.2 for n viewers located in arbitrary positions in 3-space. However, the solution can be simplified in the restricted case of two viewers, as in 5.6.3. To conserve screen resolution, a dual period barrier is introduced in 5.6.4 that can be applied at certain viewer positions, and the algorithm for doing so appears in 5.6.5. Finally, analytical results are presented in 5.6.6, with some thoughts on future optimizations using an oscillating barrier in 5.6.7.

5.6.1 Repetition of virtual lobes

Usually, the barrier period must expand by more than a factor of 2 when two viewers are present, in order to prevent a conflict between virtual lobes. To clarify this concept, consider Figure 37. Any parallax barrier AS

system, whether static or dynamic, does not project focused channels to only the COPs. Rather, these channels repeat horizontally outward in both directions, spaced at regular intervals. The period of repetition is computed by equation 10. In this discussion, the term "virtual lobe" refers to this repetition of a channel to locations other than the physical COP. Each virtual lobe repetition moving outward from the primary channel decreases in intensity.

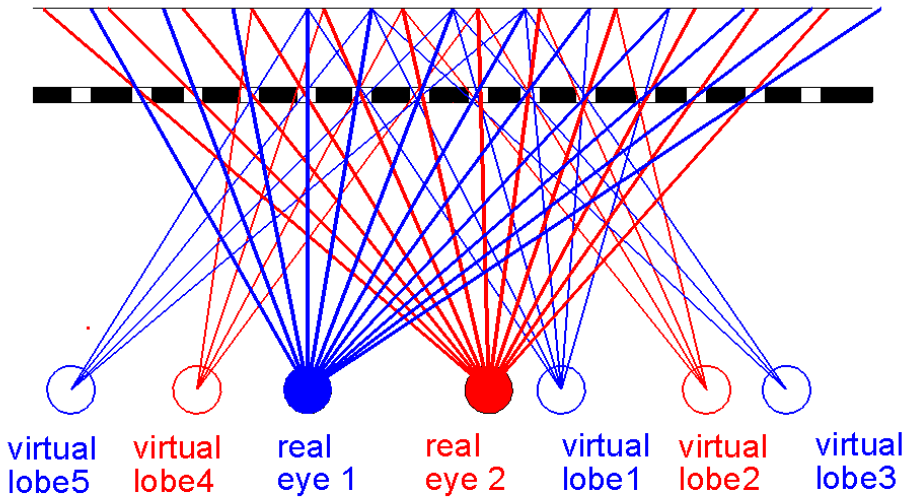


Figure 37: Repetition of primary eye channels at regularly spaced virtual lobes

$$s = p \cdot \frac{d+t}{t} \qquad (10)$$

where:

s = lobe spacing

p = barrier period

d = normal distance from eyes to front screen plane

t = optical thickness between front and rear screens

In a single viewer tracked display, the virtual lobes can be ignored for all practical purposes. They exist, but the viewer cannot see them since they occur away from the location of the eyes and so are invisible from a first person perspective. However, that is not the case with two viewers, because it is probable that one viewer will

stray into the virtual lobe of another. When this occurs, the viewer sees incorrect imagery, which can include ghosting, guard bands, pseudo-stereo, or stereo of two completely different perspectives. All of these artifacts are extremely disconcerting, and the probability of them occurring is high, because an efficient system packs channels and virtual lobes together as tightly as possible. Clearly, this is a situation that must be proactively controlled with the dynamic barrier. The next section formulates a theoretical basis for computing an optimal barrier period for any number of viewers. Later sections present optimizations specific to the two viewer situation.

5.6.2 General optimization algorithm

In the general case of n viewers at arbitrary (x,y,z) locations in space, no direct formula exists for the computation of optimal barrier period in O(1) time. Instead, the problem is cast as a minimization algorithm as follows. First, let us define the terms *conflict* and *conflict energy*. A conflict between an eye and another virtual lobe exists when the two are closer together than the interocular distance (63 mm on average, or approximately 2.5 inches). For example, in a perfectly tuned single viewer system, virtually lobes are spaced at the interocular distance; this is the nearest that they can be without causing ghosting. Moreover, the degree of conflict, called the conflict energy can be quantified as the amount that the eye and virtual lobe are nearer than this threshold distance. Mathematically, conflict energy is expressed in equation 11:

$$E_{i,j} = \begin{cases} | e - d_{i,j} |; & for\ d_{i,j} < e \\ 0; & for\ d_{i,j} >= e \end{cases} \qquad (11)$$

where:

$E_{i,j}$ = conflict energy of eye i with respect to eye j

$d_{i,j}$ = distance from eye i nearest virtual lobe from eye j

e = interocular distance

Equation 11 defines the energy of a single conflict, so the total conflict energy of the entire system is a straightforward summation of the individual conflict energies:

$$E_{total} = \sum_i \sum_j E_{i,j} \qquad (12)$$

In other words, a conflict exists whenever a virtual lobe and an actual eye position are nearer than the interocular distance. The closer they are inside of this threshold, the higher the conflict energy. Several conflicts may exist within the system for a given barrier period, and the total conflict energy is the sum of the individual conflict energies. The object of the two-viewer controller module is to set the barrier period such that no conflicts exist in the system, or the total conflict energy is below some minimal threshold at which point it is negligible. Equation 12 contains many local minima, and a linear search is the only method to find the global minimum where $E_{total} = 0$. However, the search range and step size are optimized to keep the number of iterations small; usually less than 50 are required to compute the period within sufficient accuracy.

5.6.3 Two viewer optimizations

The previous section outlined the general theory for minimizing conflicts for n viewers at arbitrary locations. Fortunately, for the special case of two viewers at a restricted subset of locations, a direct O(1) computation can be derived. Then, with a few minor extensions, this equation can be relaxed to accommodate two viewers at arbitrary locations, including different distances from the screen. This is the approach used in the final version of Dynallax, and those equations are derived in the following sub-sections. Sub-section 5.6.3.1 develops the solution for one degree of freedom, that is, two viewers separated only in their x location. Subsequently, the solution is extended to viewer separation in the y and z location, in 5.6.3.2 and 5.6.3.3, respectively.

5.6.3.1 One degree of freedom direct barrier period computation

The derivation begins with the most restrictive case for two viewers, that is, to permit only one degree of freedom, their x or horizontal displacement. Once this case is derived, extensions to two and three degrees of freedom will be added. Let us adopt the following conventions:

x = left, right location in world units, from some world origin

y = up, down location in world units, from some world origin

z = distance from front display screen, in world units

p = barrier period in world units for the single viewer case

p' = barrier period in world units for the two viewer case

d = x displacement between viewers, in world units

s = period of repetition of lobes at viewers' z distance, in world units

e = interocular distance in world units

θ = barrier tilt angle

To begin the derivation, note that in the single viewer case the system is optimal when the barrier period p causes virtual lobes to repeat in space (at the z distance of the viewer) by the spacing $2e$, where e is the interocular distance. See Figure 38:

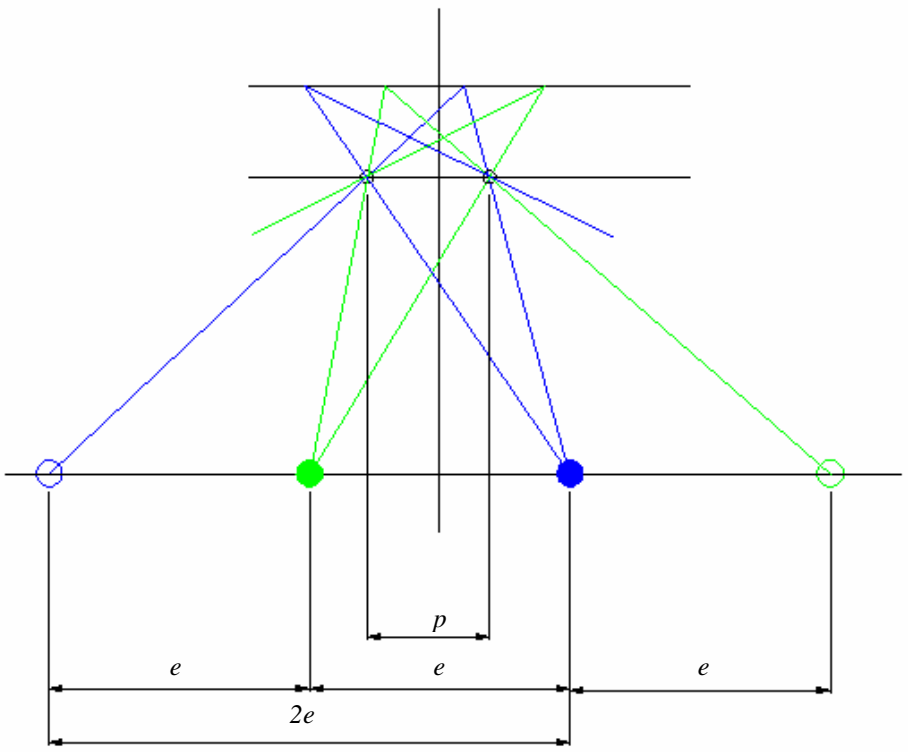


Figure 38. Optimal separation of virtual lobes

With two viewers, twice as many channels need to be interleaved on the rear screen, requiring the two viewer barrier period *p'* to be at least *2 * p*. This is the optimal two viewer condition, but is seldom attainable unless the viewers are at the optimal *x* displacement, *d*. The top image of Figure 39 demonstrates that this optimal condition occurs when $d = 2 * e$. This is the minimum that *d* can be, since the two viewers would physically collide if it were smaller. In Figure 39, solid cyan and magenta circles depict eye positions of one viewer while solid green and blue circles represent the eye locations of the second viewer. Outlined-only circles of the same color indicate virtual lobes of the same viewers, respectively.

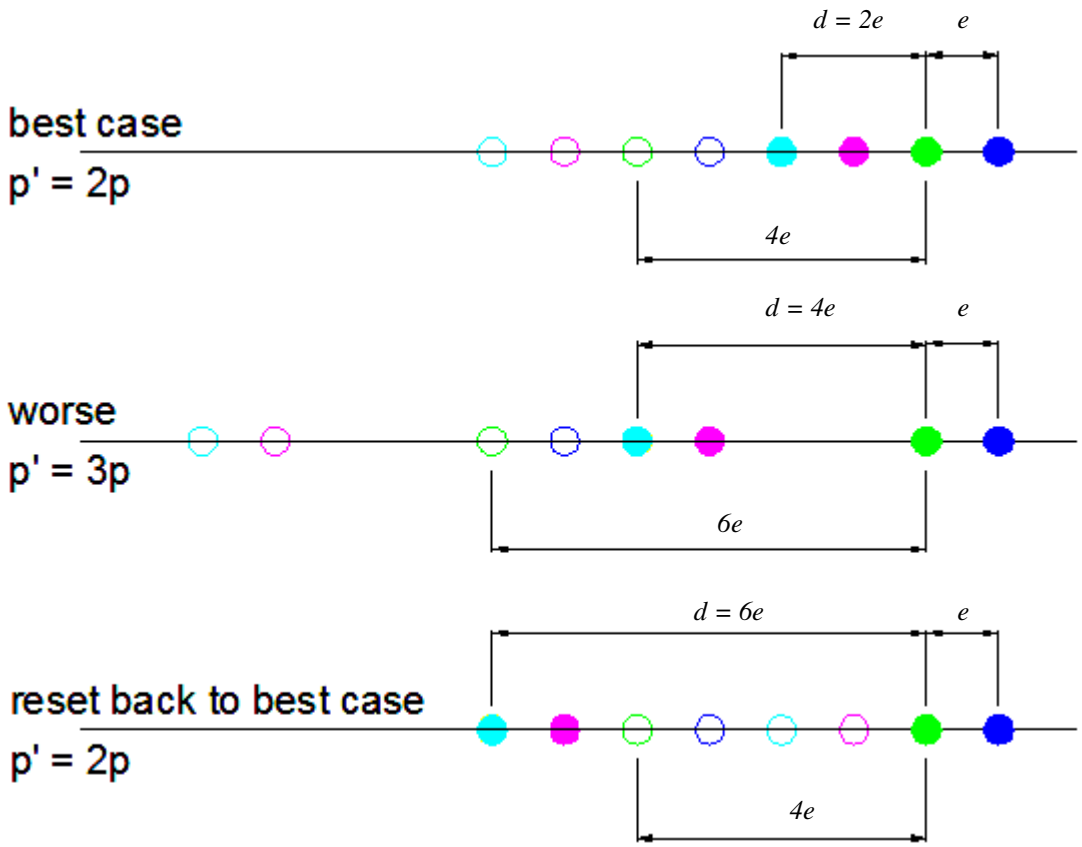


Figure 39. Several examples of viewer separation and resulting barrier period

Continuing with Figure 39, as *p'* increases proportional to *d*, efficiency decreases as shown in the middle image by the gaps between lobes. This pattern continues until a maximum value of $p' = 4p$, or as d approaches $6 * e$ in the limit from the left. At exactly $p' = 6 * e$, there is room for a second virtual lobe to exist between the two physical viewers, permitting *p'* to reset back to its optimal value of $2p$, as in the bottom image.

A pattern begins to emerge: the two viewer period *p'* is proportional to a periodic function of the *x* displacement between the two viewers. In fact, *p'* follows the sawtooth non-continuous function of *d* in equation 13 and plotted in Figure 40. Note that the single degree of freedom restriction is still in place.

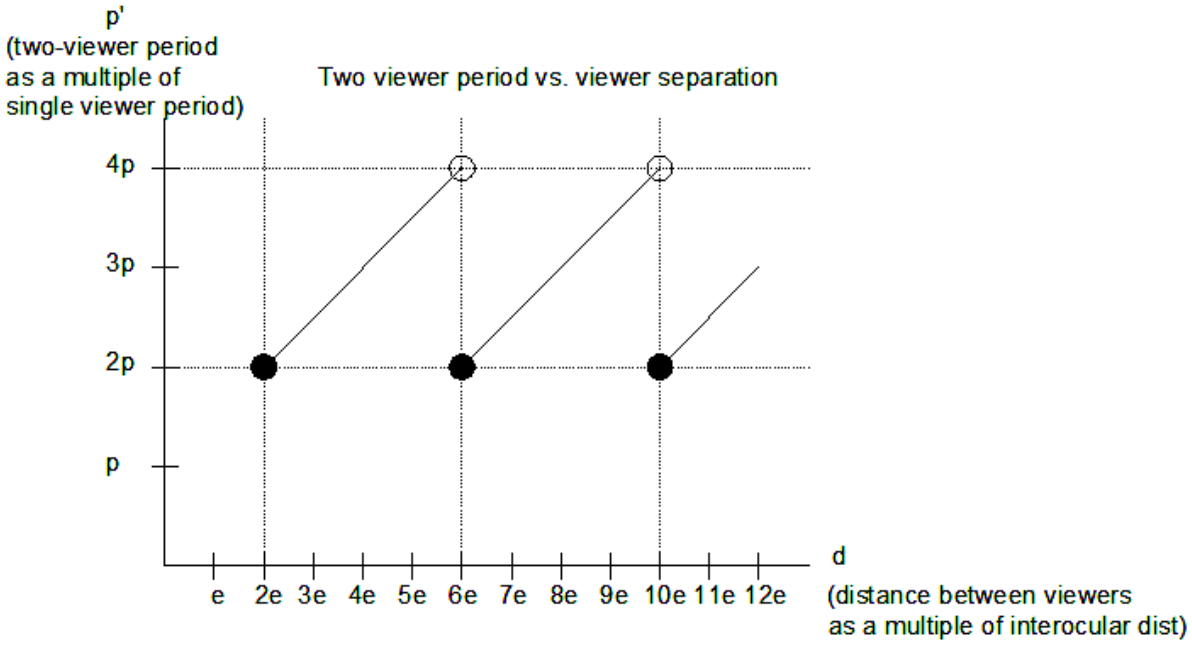


Figure 40. Sawtooth form of two-viewer barrier period function

$$p' = 2p + 2p \cdot fract\left(\frac{d - 2e}{4e}\right) \tag{13}$$

where

fract(x) ε (0.0, 1.0] and is the fractional part of a real valued x

Note that equation 13 does not include any *z* terms indicating the distance of the viewers to the screen, because this information is already encoded into the single viewer base period, *p*. The period *p* has already been optimized for view distance according to equation 5.

5.6.3.2 Extension to two degrees of freedom

Next, the derivation is extended to permit two degrees of freedom, specifically non-equal *y* positions of the two viewers. The idea is to convert the actual *x* separation of the viewers (at different *y* locations) to a theoretical *x* separation as if the two viewers were at the same *y* location. One viewer is translated to the *y* position of the other viewer along an angle equal to the barrier tilt angle θ. The new *x* position is used to compute *d*, and *p'* is computed from equation 13 as before. The transformation of one viewer to the *y* location of the other viewer and the effective value of *d* is computed for non-equal *y* locations from equation 14.

$$d = \begin{cases} |\Delta\, eye_x| + \Delta\, eye_y \cdot \tan\theta & \text{for } \Delta\, eye_y \cdot \tan\theta > 0 \\ |\Delta\, eye_x| & \text{for } \Delta\, eye_y \cdot \tan\theta < 0 \end{cases} \qquad (14)$$

Equation 14 expands the barrier period according to the first case when the difference in viewer *y* locations brings one viewer nearer to the virtual lobe of the other viewer, but does not reduce the barrier period when the opposite seemingly is true. According the second case of equation 14, the period is unchanged when it appears that it should be reduced. This is safe behavior because it is possible for virtual lobes to surround a viewer from both sides, and although reducing the barrier period may be appropriate in certain special situations, it is not in the general case.

5.6.3.3 Extension to three degrees of freedom

Finally, the last step is to permit all three degrees of freedom of viewer movement, allowing non-equal *z* locations. As in the previous extension, the strategy is to compute an equivalent value of viewer separation, *d*, as if the viewers were at the same *z* distance, and then to re-use the previous derivations from that point onward. The requisite geometry appears in Figure 41. In the left hand side, two viewers are shown at different *z* locations with their corresponding virtual lobes. The virtual lobes are focused at the same *z* distance as the physical eye locations, and the width of the lobes expands as the lobes grow out of focus both in front of and behind this *z* location. The right hand side of Figure 41 magnifies the area of interest. There, *d* represents the *x* distance

between viewers which is implicitly computed at the average of their z distance. The additional distance due to unequal z locations of the viewers is denoted by Δd in the right side of Figure 41 and computed by equation 15.

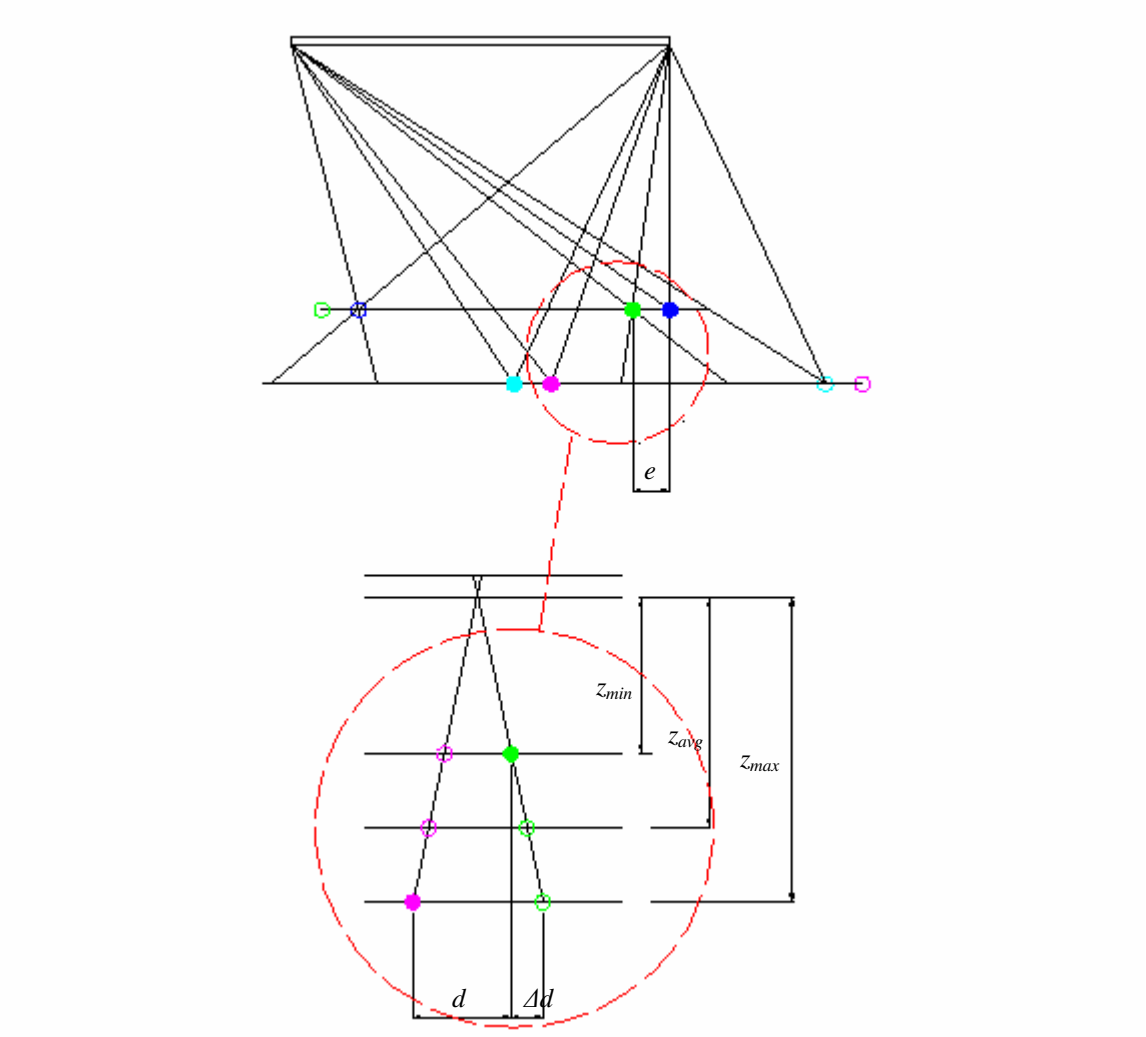


Figure 41. Geometry for non-equal z locations of two viewers

$$\Delta d = d \cdot \frac{\Delta z}{2 z_{avg}} \qquad (15)$$

The quantity Δd behaves as an addition to the viewer separation computed previously by equation 14, which forces the barrier *p'* to become even larger when viewers are not at the same distance from the display. This is true except for the case when a larger value of *d* causes the barrier *p'* to reset to its optimal value of *2p*. To prevent Δd from re-setting the barrier period, Δd is added into equation 13 after the modulo term, rather than before it, leaving equation 14 unchanged. With this change, equation 13 now becomes equation 16, and this completes the derivation of *p'* for the generic, unrestricted viewer locations in three degrees of freedom.

$$p' = 2p + 2p\left[fract\left(\frac{d-2e}{4e}\right) + d \cdot \frac{\Delta z}{2z_{avg}} \right] \qquad (16)$$

5.6.4 Dual period barrier

Figures 39 and 40 illustrate how the two viewer barrier period grows in size as two viewers become further separated in horizontal location. As the barrier expands, the system's effective resolution decreases because the barrier causes larger regions of the screen to remain black. It should be clear from the previous section that this is necessary in order to prevent conflict between virtual lobes and viewers, but the large barrier periods that can result are problematic. First, screen resolution is a precious resource in AS VR, and algorithms must make every attempt to maximize their use of it. Second, large barrier lines are visible to the viewer, occlude a majority of the VR scene, and diminish overall brightness considerably. Fortunately, one more improvement to the algorithm is possible that increases screen efficiency by 50% in some cases, depending on viewer locations. The idea is to change the barrier pattern from a simple repeating period to a dual period as shown in Figure 42. This is possible only during the second half of the sawtooth wave form in Figure 40, when the barrier period *p'* is greater than or equal to 3 times the single viewer barrier period *p*.

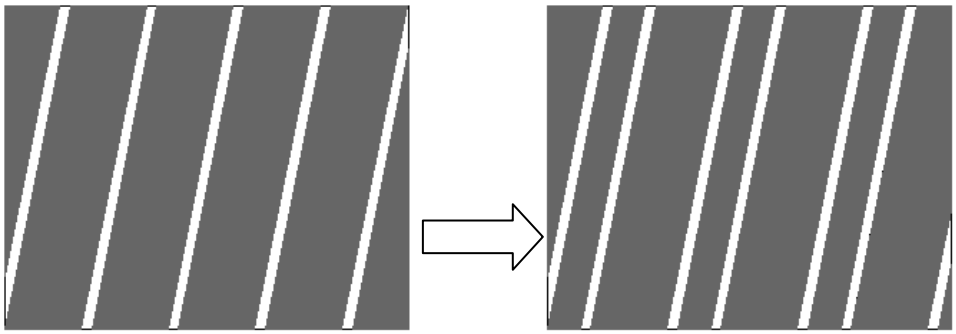


Figure 42. Dual period barrier pattern

The following progression of diagrams illustrates how and when the dual period barrier can be used, given certain viewer locations. The various color rays represent physical channels for two viewers, while black rays represent virtual lobes. In each case a top view of the system is presented, with an overall view on the right and a close-up view of the barrier area on the left. In the images, the uppermost horizontal line represents the rear screen, while the next lower horizontal line represents the front barrier screen. Viewer locations are near the bottom of the right hand image. In the left hand image, transparent slits are represented with infinitely small points, ie, duty cycle is not considered in these figures. Figure 43 is the optimal two viewer condition, where $p' = 2p$. A single period barrier is most efficient in terms of screen usage, so no further improvement is necessary. The left hand image of Figure 43 shows that the rear screen is efficiently utilized, with channels uniformly spaced.

Next, Figure 44 illustrates the situation when the viewers are slightly further apart, when $p' = 2.5p$. Note in the left hand close-up that the channels in the rear screen are less densely packed, with larger spaces between sets of four channels.

However, once the two viewer barrier period becomes greater than or equal to 3 times the single viewer barrier period ($p' >= 3p$), there is enough space to incorporate a dual barrier as shown in Figure 45. This corresponds to the second half of the sawtooth waveform in Figure 40. Screen usage returns to an optimal condition as in the left hand side of Figure 45. Notice in the left hand side that now the barrier is composed of two periods; the larger period is $3p$ and the smaller period is p, where p is the single viewer period distance.

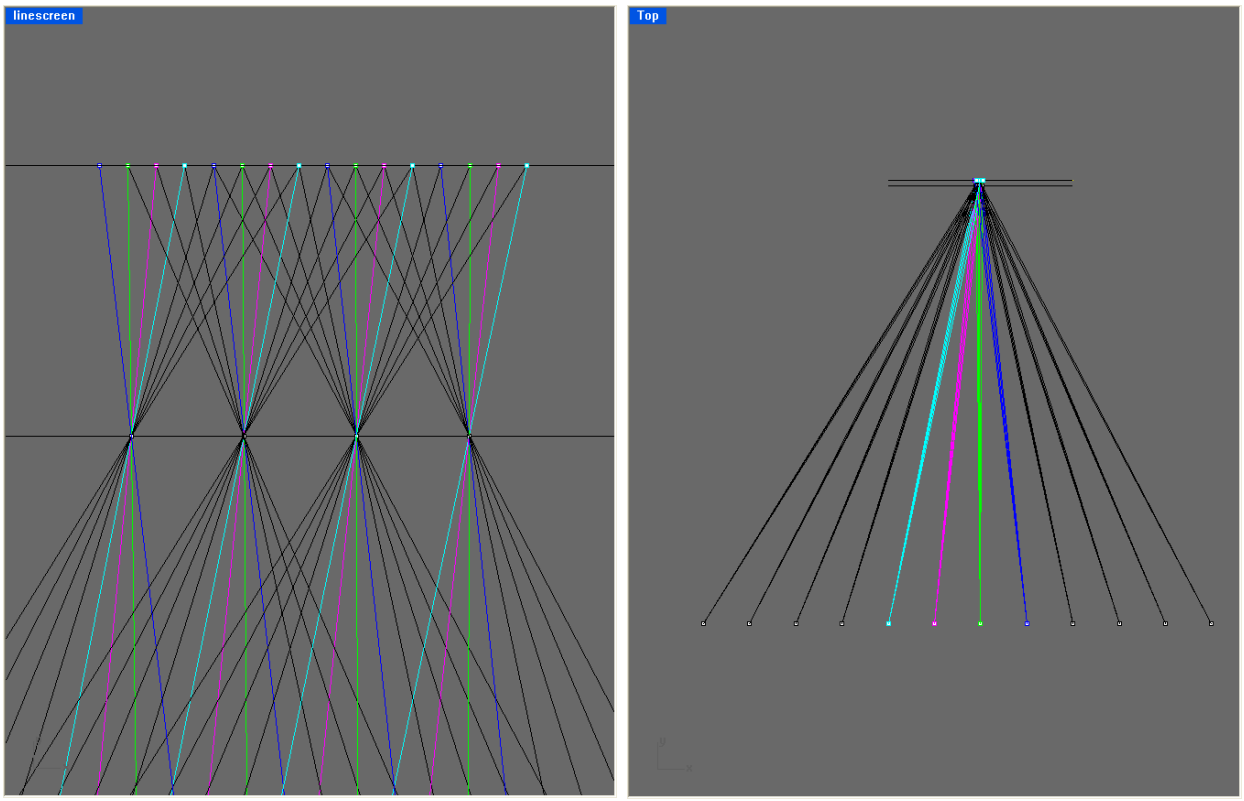


Figure 43. Optimal two viewer condition, with top view (right) and barrier close-up view(left)

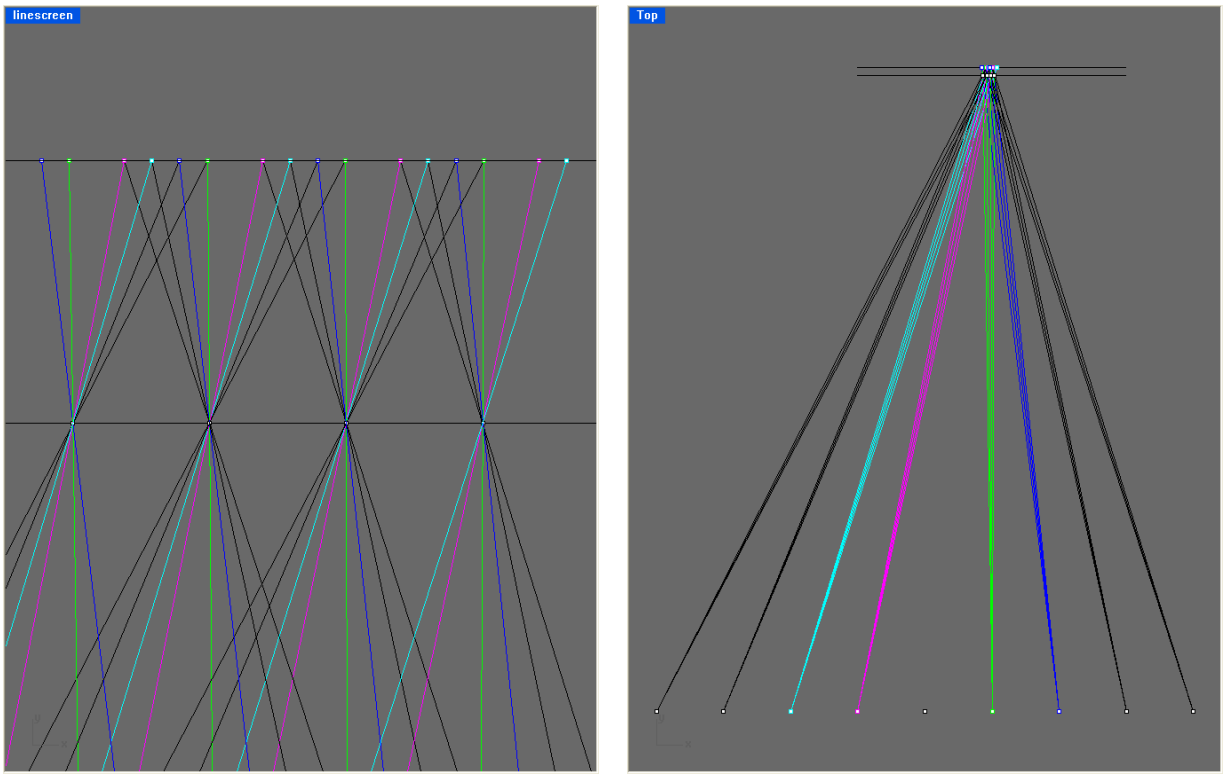


Figure 44. Non-optimal viewer condition as viewers move further apart

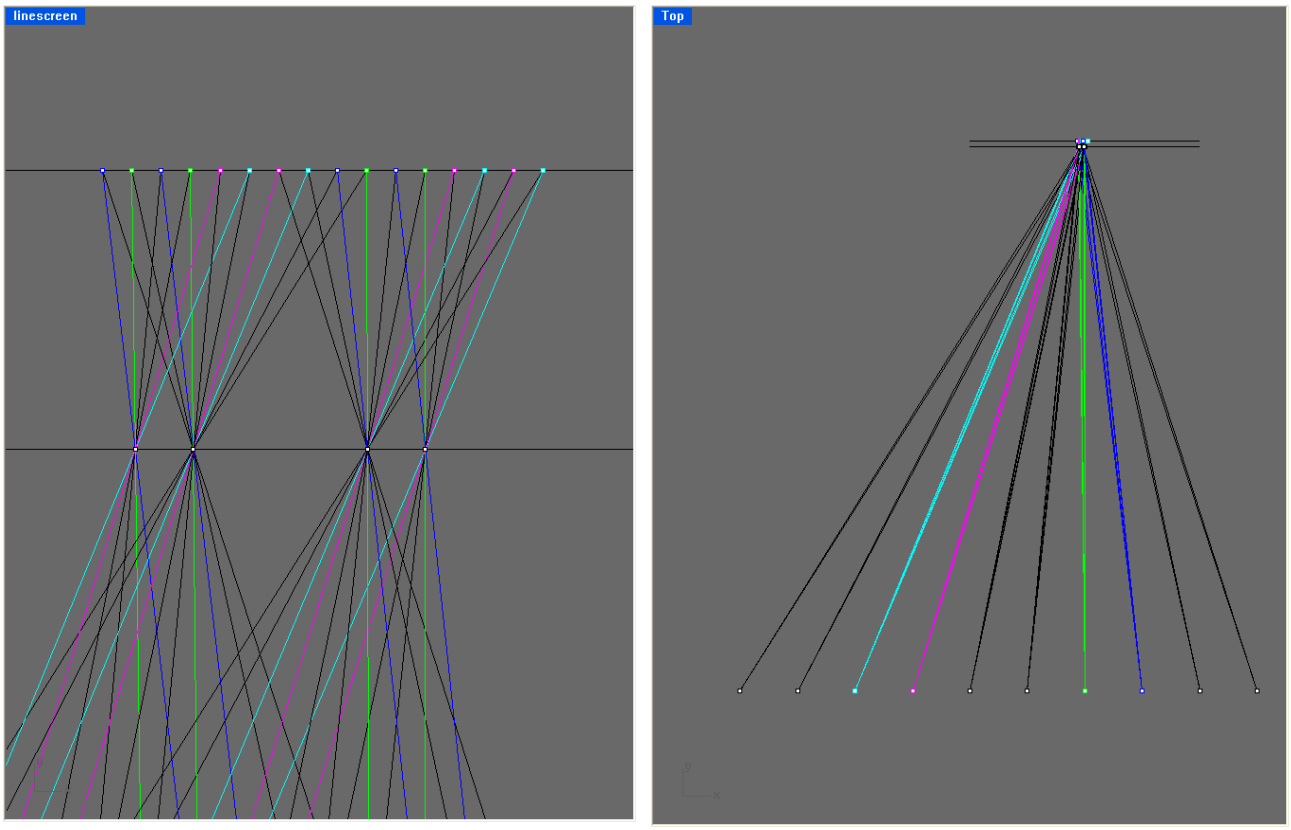


Figure 45. Optimal dual-period condition

Finally, Figure 46 shows the situation at $p' = 3.5p$. The dual period barrier remains, and like the single period barrier at $p' = 2.5p$, screen efficiency is reduced as the rear screen begins to display larger gaps between sets of channels. Beyond that, at $p' = 4p$, the entire process repeats, reverting back to the single period barrier at the optimal condition, as in $p' = 2p$.

In order to compute screen usage due to the dual period barrier, let us define the screen efficiency as the fraction of rear screen pixels that are used to compose viewable channels. Dynallax, like Varrier, relies on guard bands to separate channels. (Sandin et al., 2005) In the most efficient, optimal conditions, guard bands and channels are equidistantly arranged such that the maximum possible efficiency is .5, or in other words, half of the screen is used for guard bands. Table VII compares the efficiency of the single period barrier and the dual period barrier at various barrier periods, similar to the progression of figures presented earlier. The efficiency

gain in the far right column is computed as the ratio of the single and dual period barrier efficiencies, and represents the improvement in using a dual period barrier over a single period barrier.

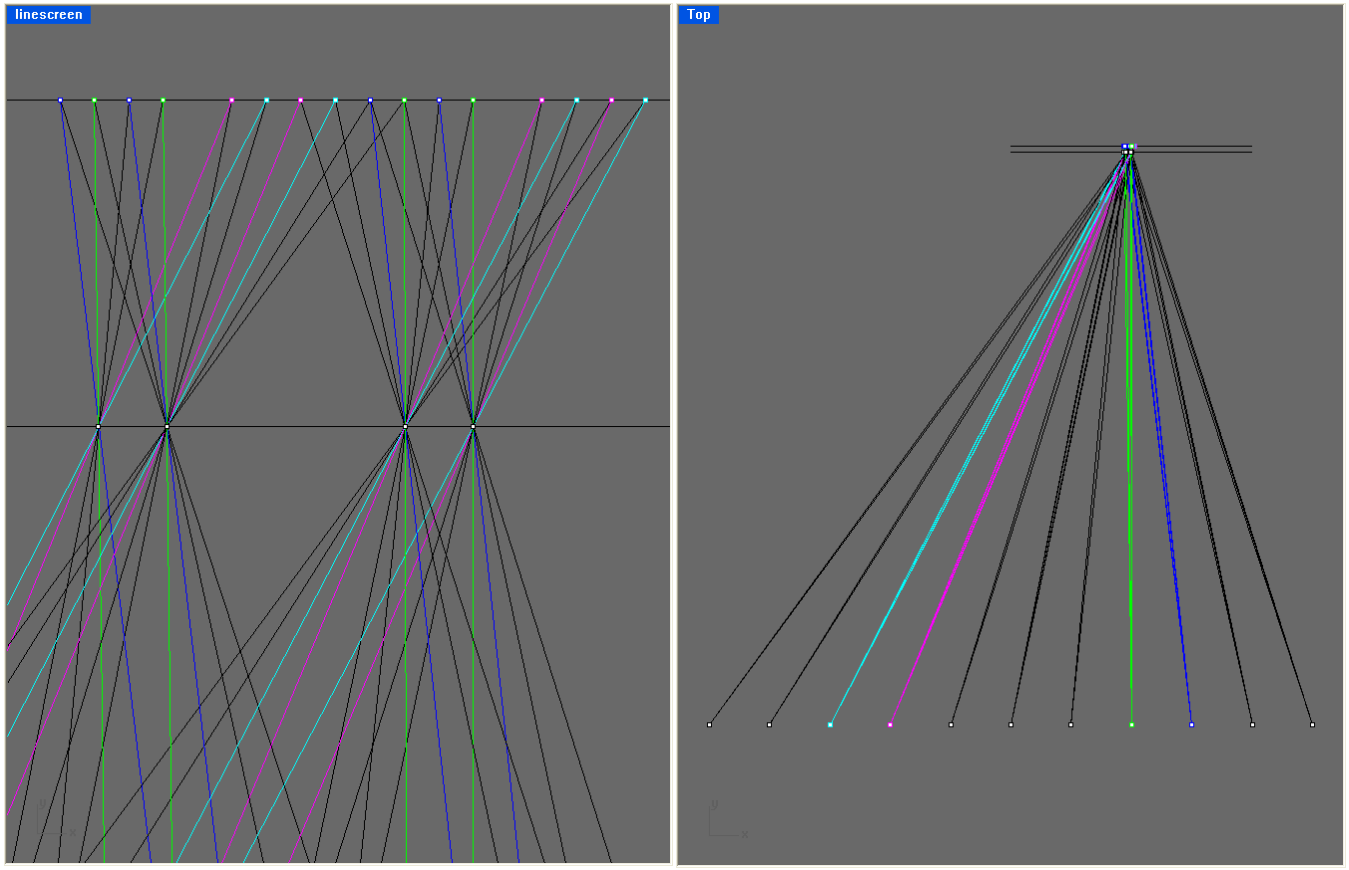


Figure 46. Non-optimal dual-period condition, with viewers further apart

TABLE VII

COMPARISON OF SINGLE PERIOD AND DOUBLE PERIOD EFFICIENCIES

p'	Single period efficiency	Dual period efficiency	Efficiency gain
2p	0.5	NA	NA
2.5p	0.42	NA	NA
3p	0.33	0.5	1.5
3.5p	0.29	0.44	1.55
3.999p	0.25	0.4	1.6
4p	0.5	NA	NA

5.6.5 Dual period algorithm

The final algorithm for the two viewer dual period barrier follows. Recall from Section 5.3.3, the derivation of the single barrier mathematics including the formation of a step function, shader implementation, and input variables between the application program and the shader program. That mathematics still applies to a large extent, and is not repeated here. Rather, the following extensions are added to that algorithm to comprise a complete solution for all cases. Five shader programs comprise the final implementation:

1. Front screen single period barrier
2. Front screen dual period barrier
3. Rear screen single-viewer (2 channels) single period barrier interleaving
4. Rear screen two-viewers (4 channels) single period barrier interleaving
5. Rear screen two-viewers (4 channels) dual period barrier interleaving

Shaders are enabled / disabled dynamically per frame, depending on whether one or two viewers are selected, and in the case of two viewers, whether p' is less than or greater than $3p$, as explained in the previous section. Earlier, the geometry of a single period step function was developed using Figures 28 and 29. Figure 47 extends that geometry to the dual period case. As before, p represents the original single viewer period, while p' is the two viewer period. The duty cycles are d and d', respectively.

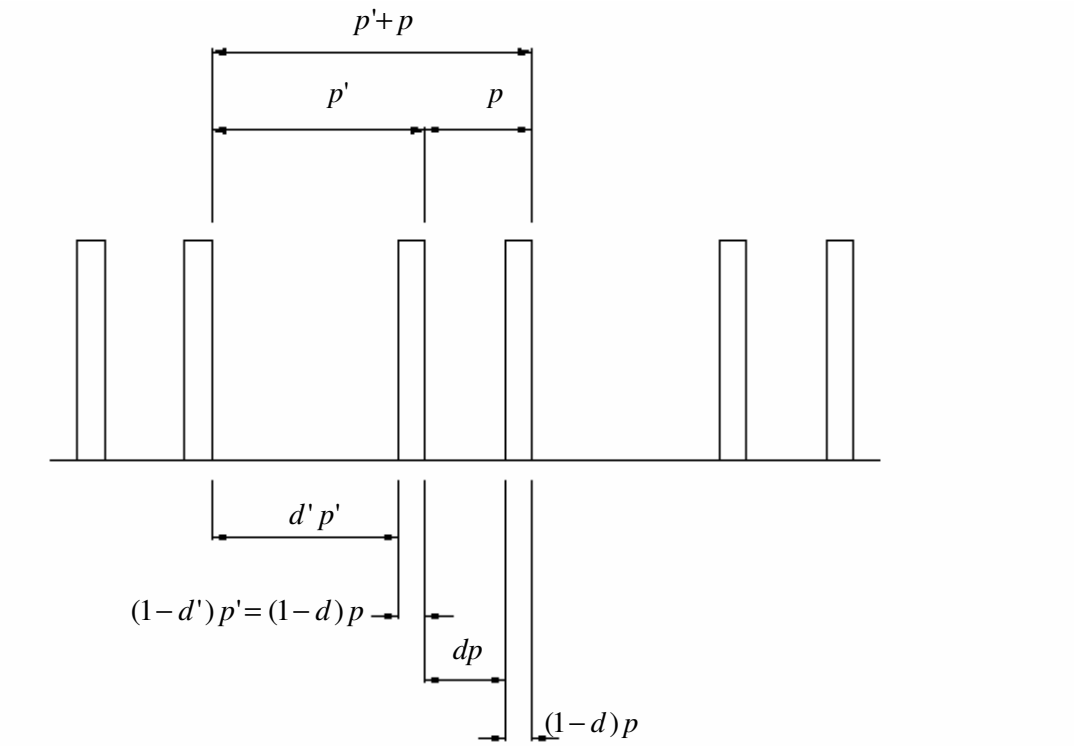


Figure 47. Dual period step function parameters

The algorithm for modulating a sub-pixel fragment in the rear screen follows from Figure 47, and is given in the pseudo-code below. First, a determination is made whether the fragment falls into the larger, p' region, or the smaller, p region. Then, a mask is computed whose result is either 0 or 1. The mask is used to multiply the incoming color fragment of the scene, making it either visible or not. The step function used in the pseudo-code is equivalent to the GLSL step(edge,x) function that returns a 1 if x >= edge, and a 0 otherwise.

```
let f = phase fraction within (p' + p)
if step( p' / ( p' + p ), f )     // f falls into p region
        mask = step( [ d, f *  ( p' + p)  – p' ] / p)
else                              // f falls into p' region
        mask = step(1 – ( 1 – d ) * p / p', f *  ( p' + p) / p')
```

5.6.6 Results

With the two-viewer controller module, two viewers can each see their own perspective of the virtual world. The controller dynamically sets the optimal barrier period given the positions of the two viewers in 3-space, and optimizes screen resolution by utilizing either a single period or dual period barrier. Figure 48 shows an actual test of the barrier period for several different viewer separations. This test was conducted prior to the development of the optimizations in Sections 5.6.3 and 5.6.4, but the form of the plot in Figure 48 resembles the sawtooth waveform of Figure 40. This helps to confirm the validity of those optimizations. The photographs underneath the graph in Figure 48 are the usual color pattern using four colors to represent the four view channels, photographed from much further back than the COP in order to see all channels simultaneously.

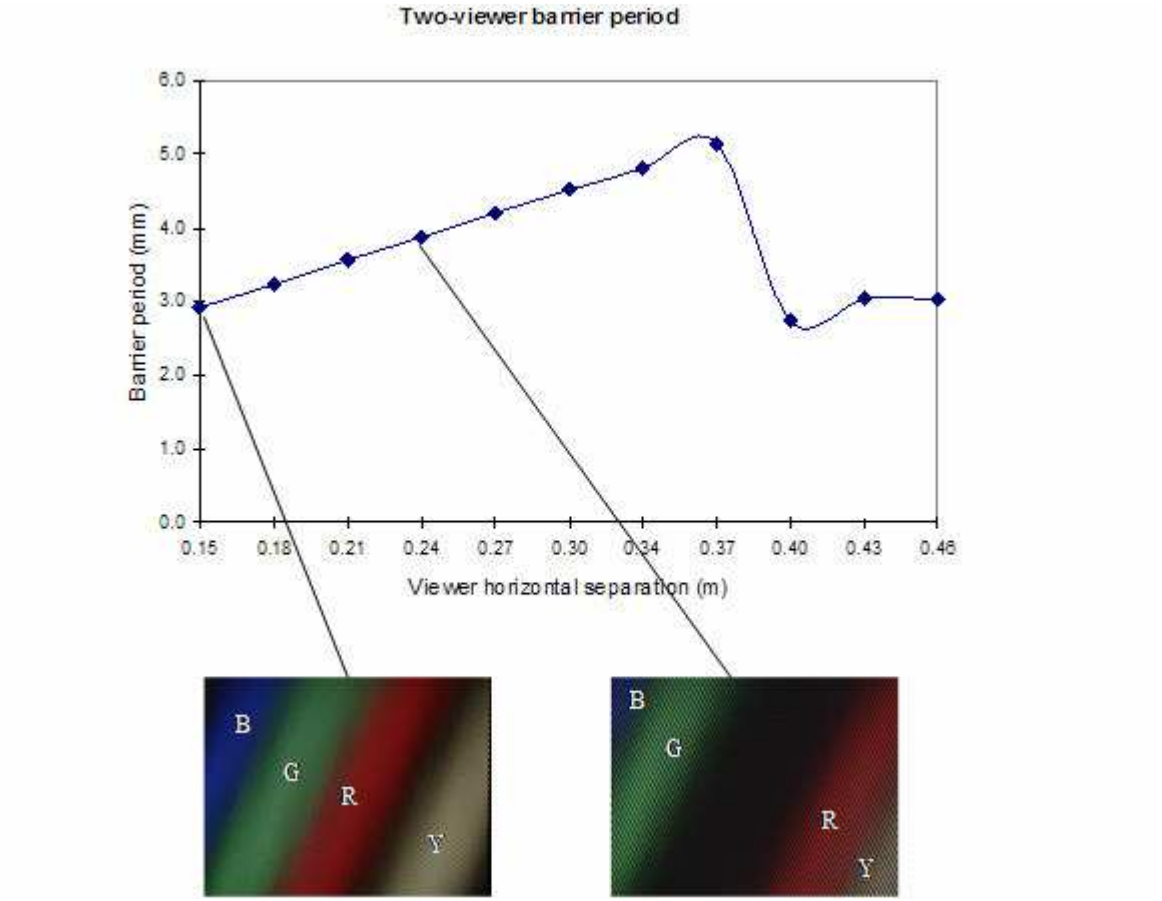


Figure 48. Output of two-viewer search algorithm tested under various viewer separations

The purpose of the next experiment is to evaluate quality in two-viewer mode. Two video cameras mounted on a jig, separated by the interocular distance, comprise the apparatus shown in Figure 49. The cross-bar pattern is rendered on the Dynallax display. In two-viewer mode, the cross bar pattern consists of vertical, horizontal, 45 degree, and -45 degree angled bars to represent the four channels. Figure 51 shows the resulting images captured by the video cameras for two fixed viewer positions. (The cameras are manually re-positioned for the second viewer location.)

In Figure 51, the left column shows the left eye view and the right column shows the right eye view. In (a) and (b), single viewer mode is enabled so that only vertical and horizontal bars are visible. The dim ghost image of the opposite bar corresponds to a ghost value of the system of approximately 7%. These images of only single viewer mode are compared with images (c) thru (f) that show vertical, horizontal, and angled bars when two-viewer mode is enabled. Images (c) and (d) correspond to the first viewer while (e) and (f) are seen by the second viewer.

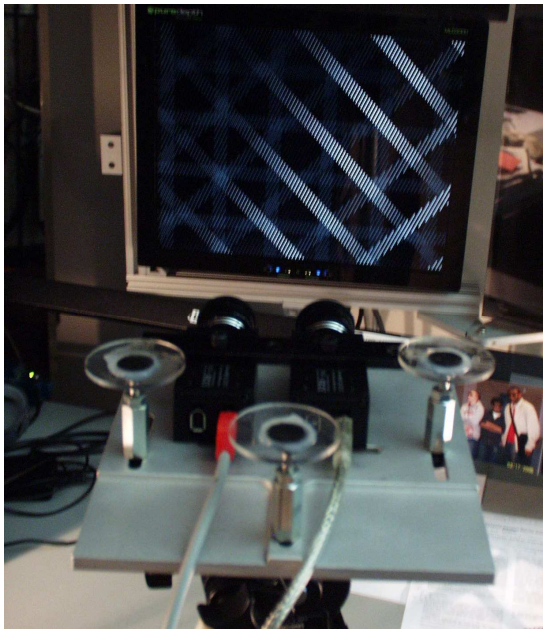


Figure 49. Experimental apparatus for capturing stereo images in Figure 50

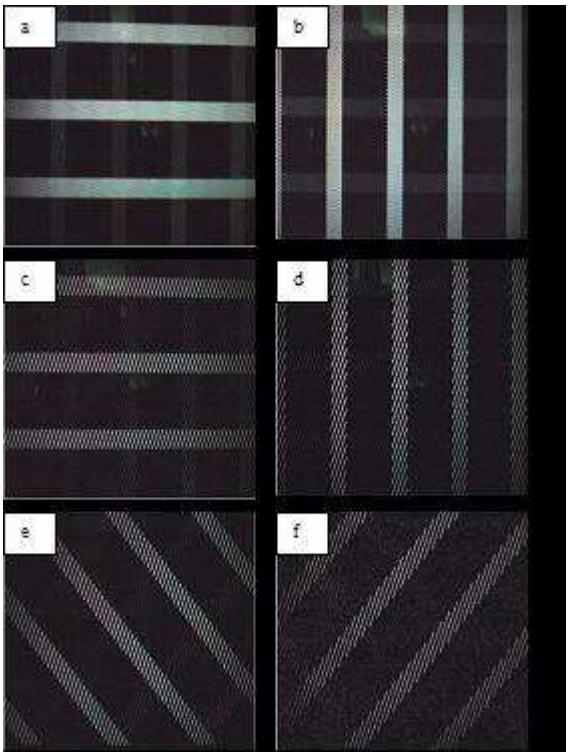


Figure 50: Left and right eye views under single and two-viewer modes captured by cameras

5.6.7 Future barrier oscillation

Despite the dual period barrier improvement, Dynallax in two-viewer mode is still a low resolution display. As Section 5.7 will show, horizontal resolution improved with a different configuration of front barrier. However, parallax barrier AS resolution is such a valuable and scarce resource that every attempt must be made to maximize its use.

Moreover, scientific research should not only consider current tools to solve problems, but needs to look one or more years down the road for future solutions to problems. The only way to keep from falling behind the curve in this rapidly changing field is to look ahead at future trends and plan for the use of technologies that do not exist yet but can be predicted with some reasonable probability.

One such trend is the increasing refresh rate in LCD display monitors. Standard LCDs currently refresh at up to 85 Hz, up from 60 Hz just a few years ago. Also, a number of manufacturers such as Hitachi, NEC, and others are now introducing flat panel LCD monitors with an 8 millisecond response time (NEC, 2007). The physical LC decay time has not changed; rather the LCs are driven by a waveform of double the frequency. In order to accomplish this, a blanking interval is introduced into each cycle, and each high voltage portion of the cycle is shorter than before. It is not clear yet whether this higher refresh frequency will one day support 120 Hz input data; currently these monitors still support input bandwidth at only up to 85 Hz.

On a related front, the ferroelectric pi-cell is a natively rapid switching LC element that has been used for many years in LC shutter glasses because of its fast response time and binary operation. Until now however, pi-cells have been limited to monochrome micro-stripe elements rather than a matrix of fully addressable color pixels. Currently, 720 Hz ferroelectric liquid-crystal-on-silicon (FLCOS) elements are beginning to appear in color matrix form, although their size is limited to approximately 1 inch x 1 inch (Lee et al., 2006). These two trends, the use of rapid LCD switching cycles and the expansion of inherently fast-switching FLCOS, lead one to believe in the future flat panel monitors that support frame rates of a least 120 Hz will be commonplace.

The background material in Chapter 2 enumerated and variety of AS systems, demonstrating that AS can be accomplished using time multiplexing, spatial multiplexing, or a combination of the two. Dynallax is a spatial-only multiplexed method, but future fast switching display technology would permit a hybrid time-space multiplexing method. The left-right position of front barrier (and corresponding rear scene) could be rapidly alternated between at least two locations at a minimum of 120 Hz, in order to be invisible to the eye. This concept is similar to the NYU display (Perlin et al., 2000, 2001), where a DLP projector engine in combination with a pi-cell barrier accomplished a similar result.

In theory, barrier oscillation would mitigate the visibility of the barrier lines, particularly for the large barrier spacing in two-viewer mode. A preliminary experiment was conducted during the early Dynallax two-viewer research whereby the visible barrier was shifted between two positions for every frame. Results were

encouraging in that barrier line visibility was reduced substantially, however flicker was quite noticeable because of the switching speed limitation of the display. The maximum 60 Hz frame rate of the display with two different barrier positions results in only 30 Hz at each position. At least twice that rate would be required; Perlin indicated that three barrier positions can further improve quality. Current LCD display limitations preclude the use of the oscillating barrier technique, suspending this thread of research in Dynallax. However, the technique can and should be re-visited when faster switching technology becomes available in a general purpose display device.

5.7 Alternative stacked display solutions

Since the front barrier is colored, light can be directed in undesired directions because a sub-pixel's light must exit through a front screen filter of the same color, which may not be collinear with the light ray from the rear sub-pixel to the eye. This situation is illustrated in Figure 51, and effectively nullifies the advantages gained by scaling the barrier down to sub-pixel size channels, since the colored front screen quantizes light transmission to single pixel resolution.

A monochrome sub-pixel LC screen is the ideal device for this algorithm, ie., an LC panel with its color filter removed. Unfortunately the color filter is an integral part of the LC layer, added during the manufacturing process. In discussions with Dan Evanicky, chief technical officer of Pure Depth (Pure Depth, 2006), it appears possible in theory to omit the color filter application step during the LC panel assembly process. With such an LC panel, 3X horizontal resolution would be gained due to the sub-pixel LC sub-structure, with optical quantization at the same sub-pixel resolution. This would be the best of both worlds. To date, a candidate front display panel has been identified by Mr. Evanicky, and discussions are ongoing to procure a prototype unit from PureDepth. However, before that can occur, various legal paperwork such as a memorandum of understanding and non-disclosure agreement needs to be agreed upon by the legal departments of the respective parties, PureDepth and the University of Illinois at Chicago. That paperwork is currently pending.

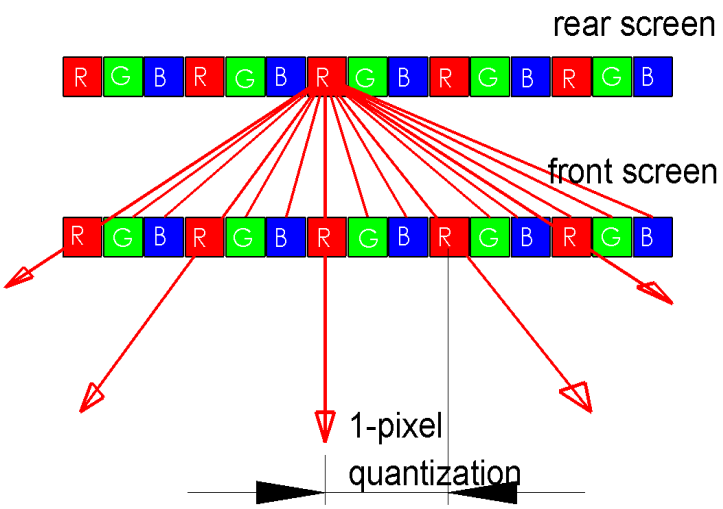


Figure 51. Quantization of light rays to nearest pixel by colored front LC panel

In the meantime, a practical solution that can lend some insight into this concept is to employ a higher resolution high contrast monochrome LC display for the front barrier, such as that used for medical imaging and diagnosis applications. The construction of such a system was introduced in Section 4.1.2. Two dissimilar LC displays increase construction complexity somewhat, for example the physical size of the LC panels may not be identical. However, there is no theoretical requirement within Dynallax that the displays be identical. Recall from Section 3.4 that the polarizer on the front screen is orthogonal the rear. In the first prototype version, this is accomplished by simply mounting the entire front LC panel orthogonal to the rear, ie, in portrait orientation instead of landscape. Likewise, electronics for driving the front panel is simply located on the outside of the display at convenient locations in terms of cable length. In a production version, these details would be refined so that both LC layers and all electronics are neatly encased inside of a single housing, as in the PureDepth display. The front screen is 2048 x 1536 pixel resolution, 21 inch diagonal, with a dot pitch of .21 mm, while the rear screen is 1600 x 1200, 20 inch diagonal, with a dot pitch of .25 mm.

Tests of this custom configuration were successful in that separation of the image into distinct channels was achieved, similar to the PureDepth display. This is a significant result, as significant as when AS first was produced in the PureDepth display. This is the first that a display of this type, custom built with dissimilar

screens including a monochrome front barrier has generated parallax barrier AS. The green-blue and cross-bar test patterns as well as a scene were tested, with comparable output to the stock display. Ghost levels were similar, in the 7% range. Brightness is diminished, but that is a function of the backlight intensity of the rear display. No attempt was made to increase the intensity of the backlight and the 250 cd / m^2 rear display is relatively dim by today's standards, even before adding the front layer.

Figures 52 and 53 document the function of the custom display in single viewer and two-viewer modes. Since the front and rear screens are oriented orthogonally to each other, correct imagery appears only where the two panels overlap, a region approximately 12 inches by 12 inches at the left of the display. In Figure 52, color calibration mode, test bar calibration mode, and normal VR scene mode are shown from left to right. As usual, images are photographed from "far away", in order to see both channels simultaneously. The custom display also can be used to display two-viewer mode, as shown in Figure 53. Here, four channels are shown in different colors, again photographed from far away. For comparison, the custom display is shown at left and the same test is performed on the stock display at right. Visual quality is comparable between the two units, as seen by the similarity in the test pattern results.

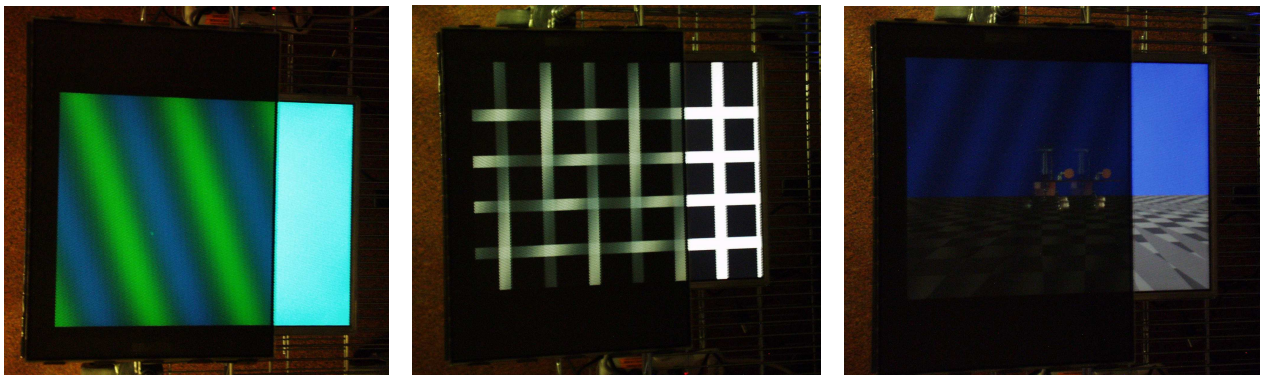


Figure 52. Tests of custom display in single viewer mode

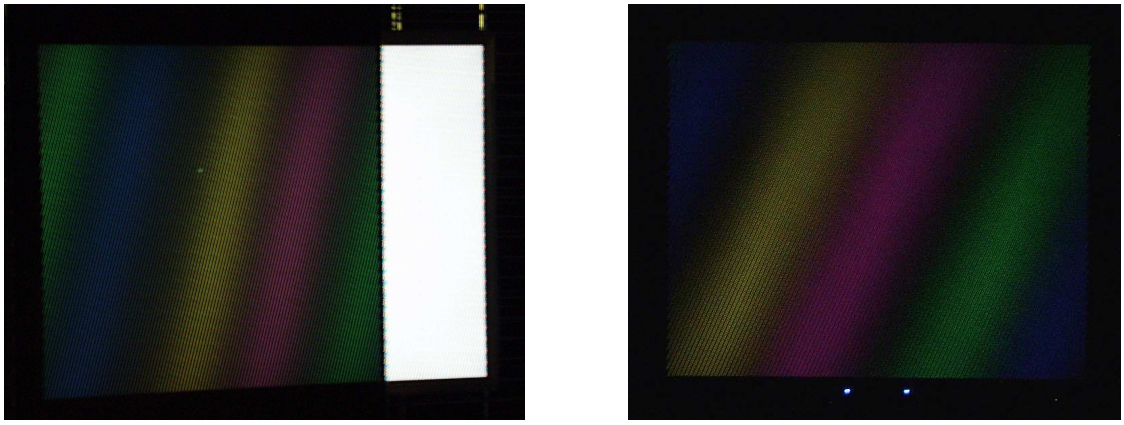


Figure 53. Test of custom display (left) compared to stock display (right) in two-viewer mode

As explained above, monochrome pixels comprise the front panel of the custom display, not monochrome sub-pixels as in the ideal solution. However, it is still instructive to compare the stock and custom display to see what if any advantage is gained in resolution. The progression of static barrier displays outlined in (Peterka et al., 2007) demonstrates that the way to improve resolution and visual acuity is decrease the barrier period. Unlike static barrier AS however, Dynallax controls the barrier period in real time according to equation 5, to maintain an optimal viewing condition. Equations 5 and 6 imply that for a given view distance, the barrier period that Dynallax selects is approximately proportional to the optical thickness. Since optical thickness is a linear function of the spacing between the two screens, it follows that the way to decrease barrier period is to physically mount the two screens closer together.

The screens cannot be arbitrarily close however. As the viewer moves further from the display, the barrier period decreases, and the minimum limit on the barrier period is one minimum resolution unit, or one front screen pixel in this case. The spacing of the screens determines where this far limit occurs. The only way to reduce the spacing between screens and maintain similar maximum view distance is to have smaller pixels. So, the advantage of the custom display over the stock display in terms of resolution should be proportional to the difference in their dot pitch, and the tests of the custom display bear this out. The comparison between the two

displays is summarized in Table VIII. The overall improvement in dot pitch and barrier period of the custom display over the stock display is approximately 25%. The various deviations from this average value in Table VIII are due to the only approximate linearity of equations 5 and 6, and to the fact that the spacing between the screens of the displays is only approximately linear in their dot pitch. These minor deviations are caused by physical mounting constraints only, not by the underlying theory.

TABLE VIII

COMPARISON OF CUSTOM AND STOCK DISPLAY

	Stock Display	***Custom Display***
Dot pitch	.25 mm (.010 inch)	.21 mm (.008 inch)
Optical thickness	.25 inch	.18 inch
Barrier period @ 12 inch distance	.10 inch	.07 inch
Barrier period @ 24 inch distance	.05 inch	.04 inch
Maximum view distance	31 inch	28 inch

5.8 Net effective resolution

Net effective resolution is a parameter that is often reported in an attempt to describe the output of parallax barrier AS systems. Unfortunately, this computation can be confusing and misleading because several approaches are possible, depending on how the term *resolution* is used. For example, it can mean anything from a simple pixel count to a physiological measure of visual acuity. Here, it is intended to count the number of minimum resolvable units (MRUs) that the system can produce. This is consistent with the approach in (Peterka et al., 2007), where the MRU is equal to one barrier period horizontally by one screen pixel vertically. The net effective resolution then is the product of the number of barrier lines and vertical pixels. In a sub-pixel barrier algorithm, the MRU is often some fraction of a pixel, and the purpose of a sub-pixel algorithm is to decrease the size and increase the number of MRUs, thereby increasing net effective resolution. However, Dynallax cannot

arbitrarily reduce barrier period since it is governed by the controller module that needs to satisfy constraints such as optimality and two-viewer conflict avoidance.

Table IX demonstrates the effect of barrier size on net effective resolution under various Dynallax conditions, including single and dual period two viewer conditions. These values are reported as “per eye”, although it is not valid to multiply by the number of eyes in an attempt to double or quadruple the net effective resolution. Compared to static barrier systems in (Peterka et al., 2007), the net resolution of Dynallax is relatively low. Although Dynallax is currently a single panel prototype and the resolution is scalable by adding panels, Table IX indicates the need for higher resolution stacked displays to be produced in the future. In particular, this is another argument for the monochrome sub-pixel front barrier. To place the net effective resolution into context, the base resolution of the display is 1280 x 1024, or approximately 1.3 Mpixel. The results of Table IX indicate that the most efficient use of this display, solely in terms of resolution, is in the single viewer mode at a 24 inch view distance, or approximately 20% efficiency. By comparison, the newest generation of Varrier displays operate at 44% efficiency, computed as the ratio of net effective resolution to base display resolution. This is to be expected since sub-pixel size channels are employed in Varrier, and it simply reiterates the earlier argument for a front barrier architecture that permits sub-pixel size channels to become a reality in Dynallax.

TABLE IX

NET EFFECTIVE RESOLUTION FOR VARIOUS BARRIER SETTINGS

Condition	**View distance**	**Net effective resolution**
1 viewer	12 inch	.13 Mpixel
1 viewer	24 inch	.26 Mpixel
2 viewers single period optimal separation	24 inch	.13 Mpixel
2 viewers single period worst-case separation	24 inch	.09 Mpixel
2 viewers dual period optimal separation	24 inch	.13 Mpixel
2 viewers dual period worst-case separation	24 inch	.07 Mpixel

6. CONCLUSIONS

A dynamic barrier is a unique and powerful test platform for studying parallax barrier AS. With a working prototype setup such as Dynallax that includes both the hardware and software for controlling the barrier in real time, a number of noteworthy findings have been discovered. Some have immediate practical advantages while others are more theoretical and can lead to future practical gains with improved hardware or other engineering improvements. The dynamic barrier is the culmination of the last six years of parallax barrier AS research by the Varrier group at EVL, of which the author has been a member for the last four years. An application for a provisional patent for Dynallax has also been filed by the author in December of 2006 with the UIC Office of Technology Management.

This chapter is divided into two sections. Section 6.1 presents findings while Section 6.2 examines ongoing and future work in the field of dynamic parallax barrier AS research.

6.1 Findings

Section 6.1 summarizes results and appraises the usefulness of each result in terms of present and future importance, both practical and theoretical. The following is a list of those findings and the appropriate sub-section number.

- Switchability between stereo, mono, and mixed modes (6.1.1)
- Elimination of color shifts (6.1.2)
- Implementation of sub-pixel barrier algorithm (6.1.3)
- Expansion of minimum view distance and produce optimal conditions over view distance range (6.1.4)
- Development of rapid view channel steering mode (6.1.5)
- Generation of 2-viewer mode (6.1.6)
- Experimentation with improved stacked display solutions (6.1.7)

6.1.1 Stereo – mono modes

This ability to switch between stereo and mono mode is extremely practical, if not groundbreaking from a theoretical standpoint. Since the front screen can become completely transparent by rendering a white background only, the display behaves as a single screen monoscopic system. A single VR scene can be rendered onto the rear screen in monoscopic mode, eliminating the need for a separate AS monitor. The same system is used for rendering AS and mono applications, or a single application that is convertible between modes. This is one solution to mitigating eye strain, a common complaint of AS users: AS is simply disabled when unneeded. Moreover, since the front screen is completely addressable, an AS barrier may occupy only a subset of the front screen, leaving the remainder transparent. In this way, the display can be spatially partitioned into both AS and mono modes simultaneously.

6.1.2 Elimination of color shifts

Section 5.2.2 demonstrated that color shifts, present in some static barrier algorithms such as Varrier 1/1, are eliminated in Dynallax by virtue of the sub-pixel structure of both the front and rear screens. This is the case even when a similar depth-buffer based single pass algorithm is tested in Dynallax. Before the advent of sub-pixel GPU rendering algorithms such as Varrier Combiner (Kooima et al., 2007) and the Dynallax sub-pixel algorithm, the only way to eliminate color shifts was to render each eye channel three times, at substantial performance cost. Now that the state of the art in both static and dynamic barrier modulation is to apply GPU sub-pixel approaches that eliminate color shifts at no additional performance cost, the prevention of color shifts by virtue of Dynallax's construction has become a redundant feature.

6.1.3 Sub-pixel barrier computational model

The Dynallax sub-pixel computational model consists of three sub-parts.

Continuous, variable, floating point barrier concept

The sub-pixel barrier concept, discussed in Section 5.3, shares a number of ideas with the Varrier Combiner model (Kooima et al., 2007). The author is indebted to Varrier collaborator Robert Kooima of EVL for his help and permission to apply those concepts within Dynallax. Specifically, the GLSL shader implementation of the step function representation of the barrier model and the RGB component-wise application of that model to the modulation of two off-screen buffers (for a single viewer) are credited to Mr. Kooima.

During the search for a suitable algorithm, the author first investigated several image-based sub-pixel algorithms, and eventually discovered that an image-based barrier, even though it is sub-pixel based, is inappropriate for this application. This is a significant result, as it led to a completely different search direction for a continuously variable floating point-based barrier. This, coupled with a GLSL implementation, coincided with Kooima's work with Varrier Combiner.

Single / dual barrier GPU shader implementation

There are several concepts that are unique to the Dynallax sub-pixel algorithm. Because of the two viewer feature, the algorithm manages twice as many view channels, modulating four channels instead of two. Since the front barrier must also be rendered, multiple fragment shader programs exist, for front and rear screens. Moreover, Section 5.6.4 formulated the motivation and mathematics for a dual period barrier, depending on the relative positions of the viewers. Ultimately, the Dynallax implementation consists of five GLSL fragment shader programs that are automatically selected as follows:

- Front screen single period barrier
- Front screen dual period barrier
- Rear screen single viewer (2 channels) single period interleaving
- Rear screen two viewers (4 channels) single period interleaving
- Rear screen two viewers (4 channels) dual period interleaving

Barrier operating equations

The third part of the barrier model consists of a set of operating equations or theorems that govern the operation of Dynallax and dynamic barrier AS in general. This set of operating principles comprise Equations 1 through 15 in Sections 5.4.2 through 5.6.3, and cover concepts such as view distance, optical thickness, barrier period, rapid steering, and two-viewer mode. A central goal of research is to understand how and why something works, in other words, what governs it. In science, the way to express the how and the why is by quantifying or modeling the behavior, in this case through a set of equations. Only when the model is complete and sound can its behavior be codified in a computer program. The easily variable nature of the dynamic barrier has afforded the ability to conduct numerous experiments by which the barrier operating principles were discovered.

6.1.4 Optimal view channels over large distances

By constantly controlling barrier period in real time, Dynallax always maintains an optimal optical condition for any view distance. In contrast, the quality of a static barrier display degrades with distance from the “sweet spot”. This optical quality is measured with ghost level. Figure 35 demonstrates that Dynallax permits closer viewing distance than previous versions of static barrier displays. This has practical importance, since viewers often prefer to move close-up to a display to see details. There is a maximum view distance in Dynallax, dictated by the distance where the transparent portion of a single barrier period is as small as a single MRU of the display. This maximum distance is the same as in a static barrier display, so nothing is lost at the far end, and view distance is gained at the near end.

This feature has a two-fold practical advantage over the static barrier display. Not only is view distance enlarged, but view quality is maintained at optimal conditions throughout the viewing range.

6.1.5 Rapid channel steering

By de-coupling the synchronization of the front and rear screens such that they each update at their fastest rates possible, and by translating the front barrier pattern as a perspective function of the viewers head translation, Dynallax provides a rapid channel steering feature. The practical applications of this principle are useful, as tracked parallax barrier AS is extremely sensitive to system latency. Since a parallax barrier system spatially multiplexes the view channels, moving faster than the system can respond results in visual stereoscopic errors such as banding, ghosting, and pseudostereoscopic vision, all very distracting.

Latency consists of frame rendering time plus communication time; rapid steering can only mitigate the frame rendering portion of the total latency. Therefore, the advantage of rapid channel steering for small scenes of a few thousand polygons is minimal, as the communication latency is the bottleneck. However, the improvement for large scenes of several hundred thousand polygons is significant. In fact, it was demonstrated that the average head velocity could be up to four times faster for a large dataset with rapid steering enabled.

6.1.6 Two tracked viewers

In the past, parallax barrier AS existed in only two variations: an untracked multi-view panoramagram that is viewable by multiple persons or a tracked two view system that is viewable by only one person. The theoretical implications of the tracked two-viewer mode are appreciable because it affords a third parallax barrier methodology. This research is the first example of two tracked viewers who each view their own independent AS perspective of a VR scene.

Since the barrier period is adjusted dynamically to avoid conflicts between viewers' real and virtual lobes, it is possible for the size of the barrier period to become up to four times that of a single viewer. The result is

diminished brightness and visual acuity, since a large portion of the screen is black for any one eye channel. To mitigate this situation, a dual-period barrier was discovered. This is also the first time such a barrier pattern has been used in parallax barrier AS, and highlights another advantage of a dynamic barrier: the ability to vary not only the size of the pattern but its character as well. As an aside, the original meaning of the term Varrier was intended to be "variable barrier". Through the years, the meaning was changed to "virtual barrier" because the variable nature of a parallax barrier was never realized. That is, until now with Dynallax.

Unfortunately, the practical usability of the two-viewer mode is still limited, mainly constrained by LCD resolution. This is especially true given the pixel quantization of colored front screen, as explained in Section 5.7 and Figure 51. Clearly, a finer LCD resolution, in particular in the front screen, results in smaller permissible barrier periods. It follows that a smaller starting single viewer barrier period produces a smaller two-viewer barrier period. The current LCD availability constrains the two-viewer mode to a theoretical contribution rather than a practical ready-to-use feature. However, the theoretical structure is in place, ready when a high resolution monochrome LCD front screen is available and cost effective.

6.1.7 Improved stacked display solution

According to officials from the manufacturer of the current stacked display used in the Dynallax prototype, a sub-pixel LCD screen with the color filter omitted is a realistic possibility. This is the ideal hardware for the front screen. While waiting for this display to materialize, the author has designed, built, and tested a custom stacked display solution. The front screen and associated electronics originated in a monochrome medical-grade LCD monitor, and the rear screen is a conventional color LCD monitor with some modifications to become a stacked display.

Tests of the custom display were successful; another first in AS research was achieved by the separation of channels using a custom display constructed from non-homogenous LCD display panels. This display had modest resolution gains of approximately 25%, corresponding to its proportionally finer pixel pitch. The overall viewing experience was comparable to the stock display, albeit dimmer since no modification was made to the

backlight intensity. However, this is still not the ideal display architecture as explained earlier. This finding demonstrates that there are no impediments to using a monochrome front screen or to using disparate screen architectures, and shows the advantage of increasing resolution of the front screen. In short, it argues for the eventual application of a sub-pixel front screen without a color filter.

6.2 Ongoing research

The procurement of a sub-pixel monochrome front panel and the manufacture of a prototype stacked display incorporating the same is an ongoing task. PureDepth, the manufacturer of the current stacked display, has agreed to provide this unit once the requisite legal documentation has been approved. This process requires the legal departments of PureDepth and UIC to agree to language in a memorandum of understanding and a non-disclosure agreement. In a related matter, the application for a provisional patent is pending so that UIC and the author can protect their intellectual property. The time frame for such legal matters historically is on the order of 6 months or longer.

In terms of research and technical details, the only impediment to the practical application of Dynallax is better display technology. Certainly the aforementioned display is vital. Other desired display technology improvements are higher contrast, higher brightness, and better saturation. These factors are constantly improving however. Since the start of this research, PureDepth has already released a newer version of a stacked display with a 2X factor of improvement in brightness and contrast. As LCD technologies continue to grow in popularity, it is expected that resolution, brightness, contrast, and saturation will continue to improve. Additionally, the evolution of faster response-time LCD technology may make a hybrid time-space multiplexing concept viable for Dynallax. This would mitigate barrier line visibility, especially during two-viewer mode.

Once an ideal display solution has been constructed and tested, scaling up to a tiled screen configuration is a straightforward extension. The current multiple process architecture is designed with scalability in mind, and there are no technical barriers to extending Dynallax to a tiled wall form factor. The porting of the Dynallax

software from a 3-node cluster to a single machine increased cost-effectiveness, especially important in a tiled context where the cost savings are multiplied by the number of tiles.

7. CITED LITERATURE

Allison, R., Harris, L., Jenkin, M., Jasiobedzka, U., and Zacher, J.: Tolerance of temporal delay in virtual environments. IEEE Virtual Reality Conference: 247, 2001.

Ascension Technology: Flock of Birds. http://www.ascension-tech.com/products/flockofbirds.php: 2005.

Adelson, S., Bentley, J., Chong, I.S., Hodges, L., and Winograd, J.: Simultaneous generation of stereoscopic views. Computer Graphics Forum 10 no. 1: 3-10, 1991.

Adelson, E.H., and Bergen, J.R.: The plenoptic function and the elements of early vision. Computation Models of Visual Processing Landy, M. and Movshon, J.A., eds., Cambridge, MIT Press, 1991.

Adelson, S. and Hodges, L.: Visible surface ray tracing of stereoscopic images. Proceedings of 30th Annual ACM Southeast Regional Conference: 148-156, 1992.

Badt, Sig Jr.: Two algorithms taking advantage of temporal coherence in ray tracing. Visual Computer 4 no. 3: 123-132, 1988.

Borse J.A. and McAllister D.F.: Real-time image-based rendering for stereo views of vegetation. Proceedings of SPIE Electronic Imaging 4660: 292-299, 2002.

Chen, S.E.: QuickTime VR – An image-based approach to virtual environment navigation. . Proceedings of ACM SIGGRAPH 1995, Computer Graphics Proceedings, Annual Conference Series: 29-38, 1995.

Cobb, J., Kessler, D., Agostinelli, J., and Waldman, M. High-resolution, autostereoscopic immersive imaging display using a monocentric optical system. Proceedings of SPIE-IS&T Electronic Imaging 5006: 92-98, 2003.

Cossairt, O., Moller, C., Travis, A., and Benton, S.: Novel view sequential display based on DMD™ technology. Proceedings of SPIE-IS&T Electronic Imaging 5291: 273-278, 2004.

Cruz-Neira, C., Sandin, D., and DeFanti, T.: Surround-screen projection-based virtual reality: The design and implementation of the CAVE. Proceedings of ACM SIGGRAPH 1993, Computer Graphics Proceedings, Annual Conference Series: 135-142, 1993.

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, no. 6: 64-72, 1992.

de Boer, D., Hiddink, M., Sluijter, M., Willemsen, S., and de Zwart, S.: Switchable lenticular based 2D / 3D displays. Proceedings of SPIE 6490: 2007.

Dodgson, N. Analysis of the viewing zone of multi-view autostereoscopic displays. Proceedings of SPIE Electronic Imaging 4660: 254-265, 2002.

Dodgson, N., Moore, J., Lang, S., Martin, G., and Canepa, P. A 50" time-multiplexed autostereoscopic display. Proceedings of SPIE Electronic Imaging 3957: 2000.

Ezell, J.D. and Hodges, L.F.: Some preliminary results on using spatial locality to speed up ray tracing of stereoscopic images. Proceedings of SPIE1256: 298-306, 1990.

Favalora, G. and Lewis, C.: Spatial 3D: The end of flat-screen thinking. Actuality Systems white paper, http://www.actuality-systems.com/index.php/actuality/content/download/283/969/file, 2003

Fernando, R., and Kilgard, J.: The CG Tutorial. Addison-Wesley, 2003.

FLTK: http://www.fltk.org, 2006.

Fuchs, H., Kedem, Z.M., and Naylor, B.F.: On visible surface generation by a priori tree structures. Computer Graphics, Proceedings of ACM SIGGRAPH 1980: 124-133, 1980.

Girado, J.: Real-Time 3d Head Position Tracker System With Stereo Cameras Using A Face Recognition Neural Network. Ph.D. dissertation, University of Illinois at Chicago, Chicago, 2004.

Girado, J., Sandin, D., Defanti, T., and Wolf L.: Real-time camera-based face detection using a modified LAMSTAR neural network system. Proceedings of SPIE-IS&T Electronic Imaging 20-24, 2003.

GLUT: http://www.opengl.org/documentation/specs/glut/spec3/spec3.html, 2006

Gortler, S., Grzeszczuk, R. Szeliski, R., and Cohen, M.: The lumigraph. Proceedings of ACM SIGGRAPH 1996, Computer Graphics Proceedings, Annual Conference Series: 43-53, 1996.

Green, S.A. and Paddon, D.J.: Exploiting coherence for multiprocessor ray tracing. IEEE Computer Graphics and Applications 9, no. 6: 1989.

Guthe, M., Balazs, A., and Klein, R.: GPU-based trimming and tessellation of NURBS and T-Spline surfaces. ACM Transactions on Graphics, Proceedings of ACM SIGGRAPH 2005 24, no. 3: 1016-1023, 2005.

Halle, M.: Multiple viewpoint rendering. Proceedings of ACM SIGGRAPH 1998, Computer Graphics Proceedings, Annual Conference Series: 243-254, 1998.

He, D., Liu, F., Pape, D., Dawe, G., and Sandin, D: 2000. Video-Based Measurement of System Latency. International Immersive Projection Technology Workshop 2000, Ames, Iowa.

Hunter, G.M.: Efficient computation and data structures for graphics. Ph.D. dissertation, Princeton University, Princeton, NJ 1978.

Igehy, H., Stoll, G., and Hanrahan, P.: The design of a parallel graphics interface. Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques: 141-150, 1998.

InterSense: InterSense IS-900 Precision Motion Tracker. http://www.isense.com/products/prec/is900/: 2005.

Jorke, H. and Fritz, M.: Infitec – A new stereoscopic visualisation tool by wavelength multiplex imaging. Proceedings Electronic Displays, Wiesbaden: 2003.

Jung, S., Park, J., Choi, H., and Lee, B.: Integral 3D Imaging that has an enhanced viewing-angle along full directions with no mechanical movement. Proceedings of SPIE-IS&T Electronic Imaging 5006: 74-82, 2003.

Kakeya, H.: Real image based autostereoscopic display using a LCD, mirrors, and lenses. Proceedings of SPIE-IS&T Electronic Imaging 5006: 99-108, 2003.

Kakeya, H., Kobe, N., and Kasano, H.: Multiview autostereoscopic display with floating real image. Proceedings of SPIE-IS&T Electronic Imaging 5291: 255-264, 2004.

Kaneko, H., Ohshima, T., Ebina, O., and Arimoto, A.: Desktop autostereoscopic display using compact LED projector. Proceedings of SPIE-IS&T Electronic Imaging 5006: 109-117, 2003.

Kay, T.L. and Kajiya, J.T.: Ray tracing complex scenes. Computer Graphics, Proceedings of ACM SIGGRAPH 1986: 269-278, 1986.

Kleinberger, P., Kleinberger, I., Goldberg, H., and Mantinband, J.Y.: A full-time, full-resolution dual stereoscopic / autostereoscopic display or rock-solid 3D on a flat screen – with glasses or without! Proceedings of SPIE-IS&T Electronic Imaging 5006: 136-144, 2003.

Kooima, R., Peterka, T., Girado, J., Ge, J., Sandin, D., and DeFanti, T.: A GPU Sub-pixel Algorithm for Autostereoscopic Virtual Reality. IEEE VR 2007 Conference Proceedings, (accepted for publication), 2007.

Langhans, K., Guill, C., Rieper, E., Oltmann, K., and Bahr, D.: SOLID FELIX: A static volume 3D-laser display. Proceedings of SPIE-IS&T Electronic Imaging 5006: 161-174, 2003.

Lee, S., Mao, C., and Johnson, K.: Fast-switching liquid-crystal-on-silicon microdisplay with framebuffer pixels and surface-mode optically compensated birefringence. Optical Engineering vol. 45, issue 12: 2006.

Leigh, J., Dawe, G., Johnson, A., and DeFanti, T.A.: ImmersaDesk 4 – A passive stereo virtual reality system using circularly polarized high resolution LCD panels. http://www.evl.uic.edu/cavern/rg/20050415_leigh/: 2005.

Levoy, M. and Hanrahan, P.: Light field rendering. Proceedings of ACM SIGGRAPH 1996, Computer Graphics Proceedings, Annual Conference Series: 31-42, 1996.

Lipton, L. and Feldman, M.: A new autostereoscopic display technology: the SynthaGram. Proceedings of SPIE Electronic Imaging 4660: 229-235, 2002.

Loukianitsa, A.,Yarovoy, A., and Kanashin, K.: Tri-Stack 3D LCD Monitor. Proceedings of SPIE-IS&T Electronic Imaging 5664: 241-246, 2005.

Lua: http://www.lua.org, 2006.

Lucente, M. and Galyean, T.: Rendering interactive holographic images. Proceedings of the 22nd annual conference on Computer graphics and interactive techniques: 387-394, 1995.

Mashitani, K., Hamagishi, G., Higashino, M., Ando, T. and Takemoto, S.: Step barrier system multi-view glass-less 3-D display. Proceedings of SPIE-IS&T Electronic Imaging 5291: 265-272, 2004.

Matusik, W. and Pfister, H.: 3D TV: A scalable system for real-time acquisition, transmission, and autostereoscopic display of dynamic scenes. ACM Transactions on Graphics, Proceedings of ACM SIGGRAPH 2004 23, no. 3: 814-824, 2004.

Microsoft: Microsoft DirectX. http://www.microsoft.com/windows/directx/default.aspx: 2005.

Miyazaki, D., Eto, T., Nishimura, Y., and Matsushita, K.: Three-dimensional volumetric display by inclined plane scanning. Proceedings of SPIE-IS&T Electronic Imaging 5006: 153-160, 2003.

Montgomery, D., Woodgate, G., Jacobs, A., Harrold, J., and Ezra, D.: Analysis of the performance of a flat panel display system convertible between 2D and autostereoscopic 3D modes. Proceedings of SPIE, 4297: 148-159, 2001.

Nvidia: http://www.nvidia.com, 2006.

MPICH: http://www-unix.mcs.anl.gov/mpi/mpich/, 2005.

NEC Inc.: http://www.necdisplay.com, 2007.

Perlin, K., Paxia, S., and Kollin, J.: An Autostereoscopic Display. Proceedings of ACM SIGGRAPH 2000, Computer Graphics Proceedings, Annual Conference Series: 319-326, 2000.

Perlin, K., Poultney, C., Kollin, J., Kristjansson, D., and Paxia, S.: Recent Advances in the NYU Autostereoscopic Display. Proceedings of SPIE 4297: 2001.

Peterka, T., Kooima, R., Girado, J., Ge, J., Sandin, D., and DeFanti, T.: Evolution of the Varrier Autostereoscopic VR Display: 2001-2007. IS&T / SPIE Electronic Imaging 2007 Conference Proceedings, (accepted for publication), 2007.

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. IEEE VR 2007 Conference Proceedings, (accepted for publication), 2007.

Pfautz, J.D.: Depth Perception in Computer Graphics. Ph.D.dissertation, University of Cambridge, Cambridge, U.K., 2002.

PureDepth: http://www.puredepth.com, 2005.

Regan, M., Miller, G., Rubin, S., and Kogelnik, C.: A real time low-latency hardware light field renderer. Proceedings of ACM SIGGRAPH 1999, Computer Graphics Proceedings, Annual Conference Series: 287-290, 1999.

Relke, I. and Riemann, G.: Special features of stereo visualization in multichannel autostereoscopic display from 4D-Vision. Proceedings of SPIE-IS&T Electronic Imaging 5006: 128-135, 2003.

Ren, J., Aggoun, A., McCormick, M.: Computer generation of integral 3D images with maximum effective viewing zone. Proceedings of SPIE-IS&T Electronic Imaging 5006: 65-73, 2003.

Rost, R.J.: OpenGL Shading Language. Redwood City, CA, Addison-Wesley Longman, 2004.

Sandin, D., Dawe, G., DeFanti, T., Leigh, J., and Verlo, A.: C-Wall (Configurable Wall). http://www.evl.uic.edu/core.php?mod=4&type=1&indi=234: 2005.

Sandin, D., Margolis, T., Dawe, G., Leigh, J., and DeFanti, T.: The Varrier auto-stereographic display. Proceedings of SPIE 4297 no. 25: 2001.

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 SIGGRAPH 2005 24, no. 3: 894-903, 2005.

Sandin, D., Sandor, E., Cunnally, W., Resch, M., DeFanti, T. and Brown, M.: Computer-generated barrier-strip autostereography. Proceedings of SPIE 1083: 65-75, 1989.

Schmidt, A. and Grasnick, A.: Multi-viewpoint autostereoscopic displays from 4D-Vision. Proceedings of SPIE Electronic Imaging 4660: 212-221, 2002.

SDL: http://www.libsdl.org, 2006.

Sherman, W. and Craig, A.: Understanding Virtual Reality. San Francisco, Morgan Kaufmann Publishers, 2003.

Shiue, L.-J., Jones, I., and Peters, J.: A realtime GPU subdivision kernel. ACM Transactions on Graphics, Proceedings of ACM SIGGRAPH 2005 24, no. 3: 1010-1015, 2005.

SPIE Stereoscopic Displays and Applications XIII conference: http://www.stereoscopic.org/2002/demo.html, 2002.

Stewart, J., Bennett, E.P., and McMillan, L.: PixelView: A view-independent graphics rendering architecture. Proceedings of ACM SIGGRAPH / EUROGRAPHICS Conference on Graphics Hardware: 75-84, 2004.

Sullivan, A.: DepthCube solid-state 3D volumetric display. Proceedings of SPIE-IS&T Electronic Imaging 5291: 279-284, 2004.

Tan, D.S., Gergle, D., Scupelli, P.G., and Pausch, R.: Physically large displays improve path integration in 3D virtual navigation tasks. Proceedings of ACM SIGCHI 2004, Conference on Human Factors in Computing Systems: 439-446, 2004.

Torborg, J. and Kajiya, J.T.: Talisman: Commodity realtime 3D graphics for the PC. Proceedings of ACM SIGGRAPH 1996, Computer Graphics Proceedings, Annual Conference Series: 353-363, 1996.

van Berkel, C.: Image Preparation for 3D-LCD. Proceedings of SPIE Stereoscopic Displays and Virtual Reality Systems VI 3639: 84-91, 1999.

van Berkel, C. and Clarke, J.A.: Characterization and Optimization of 3D-LCD Module Design. Proceedings of SPIE, Stereoscopic Displays and Virtual Reality Systems IV 3012: 179-186, 1997.

VRCO: http://www.vrco.com/CAVELib/OverviewCAVELib.html, 2006.

Ward, G.J. & Simmons, M.: The Holodeck: a parallel ray-caching rendering system. ACM Transactions on Graphics 18, no. 4: p361-398, 1999.

Whitted, T.: An improved illumination model for shaded display. Communications of the ACM 23, no. 6: 343-349, 1980.

Winnek, D.F.: Composite stereography. U.S. 3,409,351, Appl. 7 Feb 1966; 15 pp, 5 Nov 1968.

Woo, M., Neider, J., Davis, T., and Shreiner, D.: OpenGL Programming Guide. Reading, MA, Addison Wesley Longman, 1999.

Woodgate, G., Harrold, J., Jacobs, A., Moseley, R., and Ezra, D.: Flat panel autostereoscopic displays – characterisation and enhancement. Proceedings of SPIE 3957: 153-164, 2000.

Woop, S., Schmittler, J., and Slusallek, P.: RPU: A programmable ray processing unit for realtime ray tracing. ACM Transactions on Graphics, Proceedings of ACM SIGGRAPH 2005 24, no. 3: 434-444, 2005.

Wright, D.: http://dutch.phys.strath.ac.uk/CommPhys2002Exam/David_Wright/10-lcd-detail.htm, 2002.

VITA

NAME

Thomas Peterka
1251 S. Wolf Rd., Hillside IL 60162
tpeterka@evl.uic.edu

EDUCATION

University of Illinois at Chicago (UIC), Chicago, IL
Doctor of Philosophy candidate, Computer Science Engineering
Grade point average 4.0 / 4.0
Expected completion date July, 2007

University of Illinois at Chicago (UIC), Chicago, IL
Master of Science, Computer Science Engineering
Grade point average 3.8 / 4.0
December, 2003

University of Illinois at Chicago (UIC), Chicago, IL
Bachelor of Science, Computer Science Engineering
Grade point average 4.0 / 4.0
June, 1987

HONORS, AWARDS

- Best Paper Award, IEEE VR 2007 Conference, "Dynallax: Solid State Dynamic Parallax Barrier Autostereoscopic VR Display", Charlotte, NC, 2007.
- University Fellowship Award winner, UIC 2006
- First place, Industrial Advisory Board Graduate Student Poster Mini-Expo, "Multi-Camera Head Tracking for the Varrier Autostereo Display," UIC 2005.
- Bell Honor Award for highest grade point average in the College of Engineering graduating class, UIC 1987
- James Scholar full scholarship winner, UIC 1983-1987
- Honors College member, UIC 1983-1987

RESEARCH

- Autostereoscopic computer systems research scientist. Scientific visualization, scientific computing, virtual reality display systems, clustered supercomputing, high speed networking, data representation.
- Applied for patent of "Dynallax: Solid State Dynamic Parallax Barrier Autostereoscopic VR Display" through UIC's Office of Technology Management, 2007.
- Presented "Dynallax: Solid State Dynamic Parallax Barrier Autostereoscopic VR Display" at IEEE VR'07, Charlotte, 2007.
- Presented "Evolution of the Varrier Autostereoscopic VR Display: 2001-2007" at IS&T / SPIE Electronic Imaging '07, San Jose, 2007.
- Demonstrated "Personal Varrier" at Supercomputing'05, Seattle, 2005.
- Demonstrated "Personal Varrier: Autostereoscopic Virtual Reality for Distributed Scientific Visualization" at iGrid'05, San Diego, 2005.
- Attended ACM SIGGRAPH'05, Los Angeles, where "The Varrier Autostereoscopic Virtual Reality Display" was presented, 2005.

- Demonstrated 35-panel Cylindrical Varrier IEEE VR'04, Chicago, 2004.
- Developed autostereoscopic display technology and devices, including static parallax barrier (35-panel and 65-panel Cylindrical Varrier, 1-panel and 3-panel Personal Varrier, 6-panel Flat Varrier) and dynamic parallax barrier (Dynallax)

PROFESSIONAL AFFILIATIONS

- Tau Beta Pi National Engineering Honor Society
- Phi Kappa Phi National Honor Society
- Association for Computing Machinery (ACM)
- The International Society for Optical Engineering (SPIE)

PUBLICATIONS

- 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. IEEE VR 2007 Conference Proceedings, 2007.

- Peterka, T., Kooima, R., Girado, J., Ge, J., Sandin, D., and DeFanti, T.: Evolution of the Varrier Autostereoscopic VR Display: 2001-2007. IS&T / SPIE Electronic Imaging 2007 Conference Proceedings, 2007.

- Girado, J., Peterka, T., Kooima, R., Girado, J., Ge, J., Sandin, D., Johnson, A., Leigh, J., and DeFanti, T.: Real Time Neural Network-based Face Tracker for VR Displays. Trends and Issues in Tracking for Virtual Environments Workshop at the IEEE VR 2007 Conference, 2007.

- Kooima, R., Peterka, T., Girado, J., Ge, J., Sandin, D., and DeFanti, T.: A GPU Sub-pixel Algorithm for Autostereoscopic Virtual Reality. IEEE VR 2007 Conference Proceedings, 2007.

- Peterka, T., Sandin, D., Ge, J., Girado, J., Kooima, R., Leigh, J., Johnson, A., Thiebaux, M., and DeFanti, T.: Personal Varrier: Autostereoscopic Virtual Reality for Distributed Scientific Visualization. Future Generation Computing Systems, 22, 2006.

- Leigh, J., Renambot, L., Johnson, A., Jeong, B., Jagodic, R., Schwarz, N., Svistula, D., Singh, R., Aguilera, J., Wang, Xi, Vishwanath, V., Lopez, B., Sandin, D., Peterka, T., Girado, J., Kooima, R., Ge, J., Long, L., Verlo, A., DeFanti, T., Brown, M., Cox, D., Patterson, R., Dorn, P., Wefel, P., Levy, S., Talandis, J., Reitzer, J., Prudhomme, T., Coffin, T., Davis, B., Wielinga, P., Stolk, B., Koo, G.B., Kim, J., Han, S., Kim, J.W., Corrie, B., Zimmerman, T., Boulanger, P., and Garcia, M.: The Global Lambda Visualization Facility: An Ultra-High-Definition Wide-Area Visualization Collaboratory. Future Generation Computing Systems, 22, 964-971, 2006.

- Ge, J., Sandin, D., Johnson, A., Peterka, T., Kooima, R., Girado, J., and DeFanti, T.: Point-based VR Visualization for large-scale scientific datasets by real-time remote computation. Proceedings of ACM VRCIA 2006 Conference, 2006.

- Ge, J., Sandin, D., Peterka, T., Margolis, T., and DeFanti, T.: Camera based automatic calibration for the Varrier^{TM} system. Proceedings of ProCams 2005, IEEE International Workshop on Projector-Camera Systems, 2005.

- 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 SIGGRAPH 2005 24, no. 3: 894-903, 2005.

- Peterka, T.: Scientific Visualization of N-Dimensional Attainable Regions. Master's thesis, University of Illinois at Chicago, Chicago, 2003.