Designing a Horizontal Image Translation Algorithm for Use with Dynamic Stereographical Virtual Worlds

Daniel Stevenson
Department of Computer Science
University of Wisconsin-Eau Claire
Eau Claire, WI 54702
stevende@uwec.edu

Alexander Cobian
Department of Computer Sciences
University of Wisconsin-Madison
Madison, WI 53706
cobian@cs.wisc.edu

Abstract

3D imaging is gaining popularity both in Hollywood and in education, but the small-scale, low-cost systems that are useful to schools and universities face a number of challenges not present for 3D theaters. In this paper, we discuss the 3D projection system we assembled for displaying dynamic user-controlled virtual environments, the challenges we encountered due to the scope and cost of our system, and the ways that we automated the horizontal image translation process to mitigate the problems and increase the effectiveness of the system.

1. Background

There has been a sharp increase of public interest in the world of 3D imaging in recent years. Major Hollywood releases such as *Avatar*, *Alice in Wonderland*, and the upcoming *Toy Story 3* have taken full advantage of the technology, and their box-office success demonstrates that viewers are interested in the possibilities presented by stereoscopic imaging.

3D imaging is finding more applications in education, as well. Uses range from effective representation of complex scientific information (e.g., molecular structures) to accessible, memorable demonstrations of the field of computer science to prospective students. However, the 3D projection systems assembled and used by small universities must be much less expensive and physically smaller than those used in the screening of *Avatar*. These small-scale systems face a number of unique problems in producing sharp, strong 3D images. Many of these problems can be solved or mitigated through deliberate manipulation of the parallax values of the image pair (Lipton 11). In this paper, we present the basic principles of stereoscopic imaging and then describe the stereoscopic projection system we constructed and the horizontal image translation algorithm we employed to increase the effectiveness of the system.

2. Fundamentals of Stereoscopic Imaging

The basic components needed to create a single, nonmoving 3D image are two 2D images of the same scene, known as a stereo pair. A stereo pair of photographs can be made simply by taking a picture, sliding the camera several inches to the side, and taking a second picture. Though the two pictures appear nearly identical at first glance, the differences become obvious when one is made partially transparent and is superimposed over the other. Every subject captured in the photographs will be farther to the right in the image taken by the left camera and vice-versa. This distance between two corresponding points in the images is called the parallax of that point. Objects close to the camera will have high parallax, and distant objects will have low parallax.

This two-camera situation mimics that of the human eyes. The raw image data captured by a human's two eyes is essentially a stereo pair. The images are fused into the single image we are consciously aware of in the brain, but the data from the varying parallax values is not discarded. Rather, the brain remains aware of the relative parallax values at every point, and thus retains a perception of which objects are close and which are far away (Lipton 8). This perception is termed the stereo cue.

The process of 3D imaging is simply that of obtaining or generating two images of the same scene from slightly different positions and delivering them to the viewer in such a way that each image is only perceived by the respective eye. The viewer's brain then fuses the stereo pair and derives the stereo cue, resulting in the depth effect.

The simplest method by which perception of the images can be restricted to the appropriate eyes is known as the "free-viewing" method. For this, the images of the stereo pair are printed side-by-side. The viewer then tricks his/her eyes into focusing at a point either in front of or beyond the stereo pair. When the viewer's eyes are oriented such that they are pointing at corresponding points in the two images, the images will fuse and the stereo cue will be interpreted.

Figure 1: A stereo pair of photographs prepared for eyes-crossed free-viewing. (Harrison)

While free-viewing is convenient for static images that are no larger than a sheet of paper, it is impractical for even the smallest stereoscopic projection systems. Instead, one of several techniques dependent on 3D glasses must be used to restrict image data to the appropriate eyes even when the images in the stereo pair are projected directly on top of each other.

The first such technique, anaglyph imaging, involves converting the stereo pair to monochrome and projecting the images in two different colors (e.g., red and cyan). Viewers' glasses contain one lens of each color, effectively filtering out information from the incorrect channel. While this method is somewhat effective and inexpensive to implement, the loss of color is a regrettable sacrifice.

A second, similar technique involves polarizing filters and polarized 3D glasses. Each image of the stereo pair is displayed by a separate projector. Each projector has one or more polarizing filters placed in front of it in order to give light leaving that projector a

specific polarization corresponding to one of the lenses in the polarized 3D glasses worn by the users.

A third technique, the "eclipse method," partitions time rather than the visible light spectrum into left and right channels. The images of the stereo pair are projected one at a time and alternated with a very high frequency. All the viewers must wear shutter glasses, which at any given moment allow light through only one of the lenses. The glasses must alternate lens visibility at the same frequency as the projection system, and they must be synchronized perfectly, but if the alternation frequency is high enough, viewers do not notice the opaque lens.

3. System

We had three main constraints in mind when we began developing our stereoscopic projection system: scale, cost, and portability. To serve our desired purposes, the system needed to cost $2000-$3000, be capable of supporting 30-40 viewers, and be easily moved to new places and calibrated quickly.

To keep system quality high without rendering the system non-portable, we decided to employ linear polarization as our channel-isolation technique. Our system thus incorporates two identical projectors, each with a different polarizing filter suspended in front of its lens. It is necessary that the images of the stereo pair be projected in precisely the same location on the projection surface, so we mounted the projectors in a vertical rack to maximize the overlap of the two projection ranges.

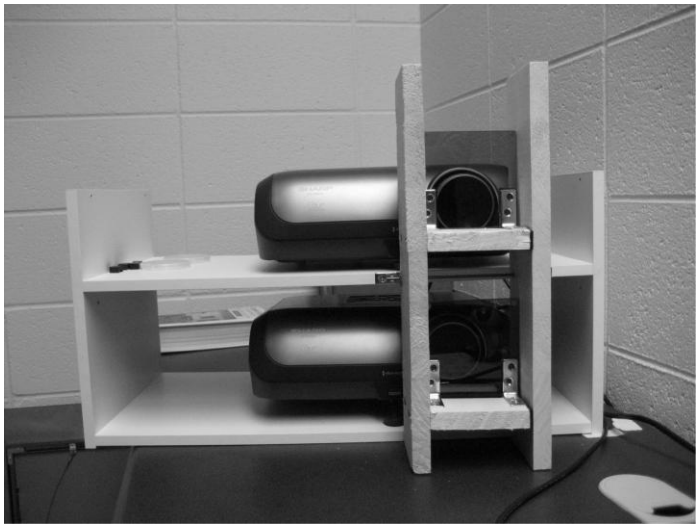

Figure 2: Our stereoscopic projection system

Despite the close projector proximity afforded by the rack, there remains a ~1.5 foot vertical parallax between the images thrown by each projector. This remaining parallax has to be eliminated using other methods. Our original solution to the vertical parallax problem was to incline the bottom projector upwards at an angle such that the center of its projection range matched that of the top projector and then use the projector's built-in keystone correction functionality to distort the image back into a rectangle. However, we discovered that it was only possible to calibrate the resulting system to within several centimeters of zero parallax in a reasonable amount of time. We now simply have each projector display its image only in the portion of its projection range which intersects the other projector's projection range. The loss of overall projection range is unfortunate, but the new solution permits us to quickly calibrate the system to within 1 millimeter of zero parallax.

The images thrown by the projectors are passed through 45° and 135° linear polarization filters (one per projector) and reflect off a silver screen. Using a highly reflective silver screen is important for our implementation because less reflective surfaces would scatter the polarized light, allowing data to leak between the left and right channel. The reflected light is then observed by viewers who are wearing 3D glasses with polarized lenses.

Initially, we used the system to display photographic stereo pairs without the need for free-viewing. Once we were certain of the effectiveness of the system, we moved on to the projection of virtual worlds in 3D. Dynamic, user-controlled worlds dramatically increase both the utility and the entertainment value of the projection system. We chose to use Java3D to construct our worlds due to built-in support for stereoscopic virtual reality applications. Both virtual cameras are bound to an invisible entity that can be manipulated via the keyboard. Allowing additional components of the scene to be manually manipulated is a simple matter as well.

4. Challenges

Unfortunately, stereoscopic projection systems of this scale and price range face several significant problems that aren't major issues in 3D theaters. Two of the most important are crosstalk and the breakdown of accommodation and convergence.

Crosstalk is a term which refers to any data which leaks between the left and right channel at some point before the user perceives it. There are several opportunities for crosstalk to occur: the light may not be perfectly polarized when it passes through the projector filter, the light may not reflect perfectly off the silver screen, or the polarized lenses of the 3D glasses may fail to prevent unmatched data from entering. If the

crosstalk received is somewhat similar to the data which was intended to reach that eye (i.e., when the viewer is focused on a low-parallax point), the extra data will likely be fused into the image, though the stereo cue may be adversely affected. If, however, the crosstalk differs significantly from the correct data for that eye, the user will experience "ghosting" (additional copies of the image floating alongside the primary copy), ruining the 3D effect.

Most 3D theaters reduce crosstalk through the use of expensive circular polarizing filters or other financially prohibitive light spectrum separation techniques. While completely eliminating crosstalk from an inexpensive, small-scale system is difficult, ghosting can be minimized if care is taken to ensure that most of the objects the viewers focus upon are at low parallax settings.

The breakdown of accommodation and convergence is a conflict of two depth cues that can result in poor stereo image quality or ocular pain (McVeigh 307). Convergence refers to the degree to which the eyes are oriented inward. (The closer an object, the more the eyes must converge to focus on it.) Accommodation refers to physical movement *within* the eye that serves the same purpose to the eye as focusing a lens does to a camera. Since accommodation and convergence are both a function of the depth of the object the viewer is looking at, they directly correspond at all times other than when doing stereoscopic viewing. When the viewer focuses on a 3D image, however, the eyes must accommodate to the projection surface while they converge upon the point where the object *appears* to be, whether that be behind, at, or in front of the projection surface. If the object spends too much time virtually in front of the projection surface, the unusual combination of accommodation and convergence values will make the viewer uncomfortable.

The negative effects of both crosstalk and the breakdown of accommodation and convergence scale with the parallax of the point the viewer is focused on. Thus, one way to minimize these problems is to ensure that the viewers' focus is normally on a point with a low parallax value. The viewers still perceive the entire scene as having depth, but breakdown does not occur since the eyes are converged at the depth of the actual projection surface. Furthermore, ghosting goes largely unnoticed because it is restricted to the peripheral vision.

We have suggested that when cameras (physical or virtual) are used to capture a stereo pair, they are placed parallel to each other, focused at some point infinitely far away. It is possible to instead orient each camera inward to face a focal point in the scene. Since this focal point will be captured in the center of each image in the stereo pair, it will have zero parallax. This is known as the "toe-in" method of parallax management.

While using the toe-in method to converge the cameras on the focal point will ensure that it has zero parallax (and thus will cause minimal ghosting and accommodation/convergence breakdown problems for the viewers), it is not recommended due to other problems it introduces (Lipton 35). Foremost among these are the introduction of vertical parallax to the image around the edges and the divergence of the images at depths beyond the focal point.

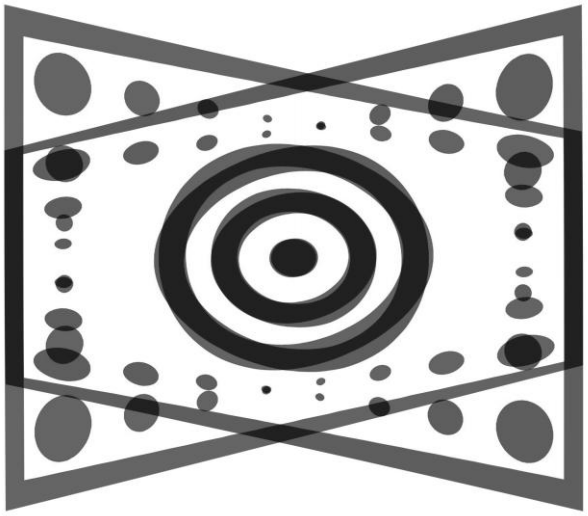

Figure 3: An exaggerated representation of superimposed pictures of the same subject taken using the toe-in method. Note the vertical distance between like points in all places except the center.

5. Automatic Horizontal Image Translation

Horizontal image translation (HIT) is a method for altering the parallax values of a stereo pair without introducing any distortion or capturing different portions of the scene beyond the focal point. The cameras are placed in parallel alignment (focused on an infinitely distant point). The resulting images are then superimposed and slid horizontally until the desired focal point has zero parallax. The parallax of every other point in the stereo pair will have been altered by the same amount, so relative depth of all objects in the scene is preserved.

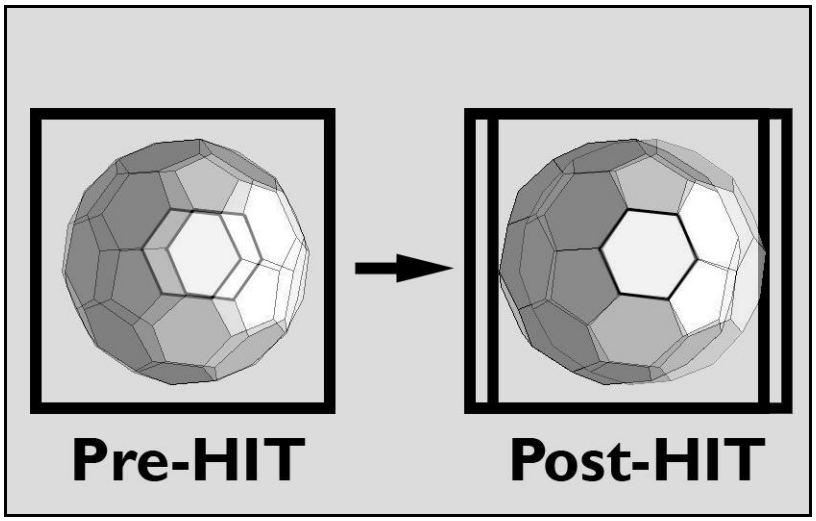

Figure 4: A superimposed stereo pair before and after horizontal image translation. The bold hexagon is the focal point of the scene.

Using HIT to optimize the parallax values of a single static stereo pair displayed by a stereoscopic projection system is a trivial task: a human (not wearing 3D glasses) need only adjust the position at which each image is displayed until the intended focal points in the left and right channel are overlapping. However, this process cannot be applied to dynamic, user-controlled worlds, as the depth of the focal point is in constant user-controlled (and hence unpredictable) flux.

The size of the optimal shift (measured in pixels) is a function of several different aspects of the virtual world and projection system. However, only one of these aspects is expected to change in a given projection session: the depth of the focal point relative to the virtual camera plane. Since this single value is not difficult to calculate, our first automatic HIT-performing system was based on a lookup table. We placed a virtual object at various depths, performed manual HIT until the focal point had zero parallax, and recorded the optimal HIT distances at these depths into the table. We could then perform optimal HIT automatically by simply determining the depth of the focal point, finding the closest depth in the lookup table, and performing the listed amount of HIT.

We later decided to find a general solution for the problem of HIT optimization, geometrically relating the relative distances between the cameras and focal object in virtual space to the relative distances of their representations in the captured stereo pair. Since the cameras and object both exist in virtual space but the stereo pair must be translated in terms of the projectors' space, we had to develop a way to convert virtual distances to pixels. We found that the number of pixels that the images of the stereo pair had to be shifted could be found via the following expression:

$$p\left(\frac{a}{2\tan\left(\frac{\theta}{2}\right)(d-r)}\right)$$

d is the depth of the center of the object from the plane of the virtual cameras and *e* is the radius of the object, so (*d* - *r*) is the depth from the virtual cameras to the focal point. θ is the field of view of each virtual camera, so $2\tan\left(\frac{\theta}{2}\right)$ gives the width of the visible virtual world over (*d* - *r*) for every depth (*d* - *r*). Taken together, $2\tan\left(\frac{\theta}{2}\right)(d-r)$ is the width of the visible virtual world at the depth of the focal point. *a* is the interaxial distance (the virtual distance between the two cameras), which is also the width of the visible virtual world at depth 0. Thus, $\left(\frac{a}{2\tan\left(\frac{\theta}{2}\right)(d-r)}\right)$ provides us with a number in the range [0,1] which represents the portion of the width of the visible virtual world at depth (*d* - *r*) for each camera which is not also captured by the other camera. (I.e., it tells us how far apart the left and right images of a point at that depth will be in terms of the overall width of the visible virtual world at that depth.) We can then multiply this by *p*, the horizontal resolution of the display, to determine the total number of pixels that the two images must be shifted outward to make every point at that virtual depth have a parallax value of 0.

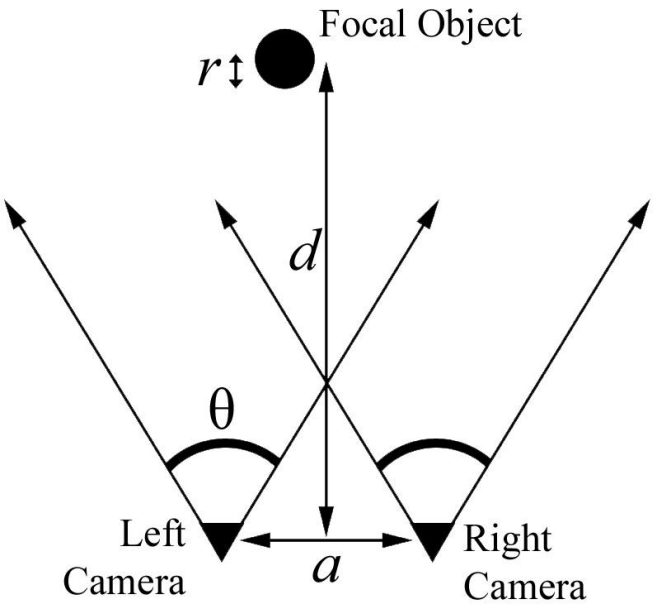

Figure 5: A diagram of the aspects of virtual space that factor into automatic HIT

To maximize the usefulness of this process, a few issues with HIT need to be dealt with. Whenever HIT is performed, the width of the resulting image is less than the width of the individual images in the stereo pair. Since the optimal amount of HIT to perform in a

dynamic virtual world context is constantly changing, the captured images of the virtual world need to be wider than the portion that will be displayed to prevent the displayed image from growing and shrinking as the cameras move nearer and farer from the focal point. Essentially, if the horizontal resolution of the display is p pixels, each virtual camera must produce an image $(p + \varepsilon)$ pixels wide. If t is the number of pixels which the images must be translated to perform the desired HIT (as determined by the above expression), then the remaining overlapping image will be $(p + \varepsilon - t)$ pixels wide. Of this, $(\varepsilon - t)$ columns of pixels must be trimmed from each image. This assures that the final image sent to the display will always be p pixels wide as long as $t \leq \varepsilon$ (a constraint that must be enforced).

The system can also be improved by introducing an acceleration constraint that allows the pixels of HIT performed to slightly "lag behind" those suggested by the expression above, in the same way that a trailing camera's movement in a monoscopic 3D virtual world will often lag behind the user-controlled entity. Without acceleration constraints, a viewer zooming in toward a focal point may find that the stereo effect is disturbed due to the parallax value of the focal point remaining constant during the zoom. This means the stereo cue suggests to the viewer that his/her position isn't changing relative to the focal point, in conflict with other monoscopic depth cues such as the increasing size of the object. Allowing some parallax variance at the focal point during the zoom helps to prevent this conflict of cues.

A third consideration, ignored in the above description of automatic HIT, is that determining the focal point in a user-controlled dynamic virtual world is itself nontrivial. The best way to determine the focal point may depend partially on the nature of the scene, but a simple recommendation would be the closest point of the closest object to the cameras that is within both cameras' field of view. However, a perfect system would take into account other characteristics of each object such as its speed, size, occlusion of/by other objects, how recently it entered the field of view, and even the importance of the object to the scene from a thematic standpoint.

6. Conclusions

We believe that the addition of automatic horizontal image translation to our stereoscopic projection system has significantly improved the quality of the perceived image. However, quantitative data supporting improvement in stereoscopic imaging is difficult to obtain. Nothing can be recorded or measured, and due to the important role of the human brain in creating the final 3D image, all results are subjective. However, a thorough test of the system could involve displaying a number of scenes (both HIT-

optimized and unaltered) to a varied test audience and polling them on which images they find most effective. It is our intent to perform this sort of test on the system in the future.

References

Faugeras, O.D.. *Three-dimensional Computer Vision: A Geometric Viewpoint*. London: The MIT Press, 1993. Print.

Forsyth, David A., and Jean Ponce. *Computer Vision: A Modern Approach*. US Ed. Alexandria, VA: Prentice Hall, 2003. Print.

Haralick, Robert M., and Linda G. Shapiro. *Computer and Robot Vision, Vol. 1*. New York: Addison-Wesley, 1991. Print.

Harrison, JJ. *Asiatic Hybrid Lilium Stereogram Flipped*. N.d. Wikimedia Commons. Web. 19 Mar. 2010.

Lipton, Lenny. *The StereoGraphics Developers' Handbook*. San Rafael, CA: The StereoGraphics Corporation, 1997. Print.

McVeigh, J. S., M. W. Siegel, and A. G. Jordan. "Algorithm for automated eye strain reduction in real stereoscopic images and sequences." *Human Vision and Electronic Imaging* (February 1996): 307-316. Print.