Balancing Fusion, Image Depth and Distortion in Stereoscopic Head-Tracked Displays

Zachary Wartell, Larry F. Hodges, William Ribarsky

GVU Center, College of Computing, Georgia Institute of Technology

Revision 1.1

REFERENCE: Zachary Wartell, Larry F. Hodges, William Ribarsky. Balancing Fusion, Image Depth and Distortion in Stereoscopic Head-Tracked Displays. In *SIGGRAPH 99 Conference Proceedings*, Annual Conference Series, pages 351-357. ACM SIGGRAPH, Addison Wesley, August 1999.

REVISION:

1.1 : In discussion of OpenGL implementation the “$[\alpha]_{view} = T^{-1} \bullet \alpha \bullet T$” included an implicit scale factor. For clarity I replaced this with “$[\alpha]_{view} = S^{-1} \bullet T^{-1} \bullet \alpha \bullet T \bullet S$” in order to make the scale factor’s necessity clearer.

Balancing Fusion, Image Depth and Distortion in Stereoscopic Head-Tracked Displays

Zachary Wartell[1], Larry F. Hodges[2], William Ribarsky[3]

GVU Center, College of Computing, Georgia Institute of Technology

Abstract

Stereoscopic display is a fundamental part of virtual reality HMD systems and HTD (head-tracked display) systems such as the virtual workbench and the CAVE. A common practice in stereoscopic systems is deliberate incorrect modeling of user eye separation. Underestimating eye separation is frequently necessary for the human visual system to fuse stereo image pairs into single 3D images, while overestimating eye separation enhances image depth. Unfortunately, false eye separation modeling also distorts the perceived 3D image in undesirable ways. This paper makes three fundamental contributions to understanding and controlling this stereo distortion. (1) We analyze the distortion using a new analytic description. This analysis shows that even with *perfect* head tracking, a user will perceive virtual objects to warp and shift as she moves her head. (2) We present a new technique for counteracting the shearing component of the distortion. (3) We present improved methods for managing image fusion problems for distant objects and for enhancing the depth of flat scenes.

CR Categories and Subject Descriptions: I.3.7 [Computer Graphics] Three-Dimensional Graphics and Realism - Virtual Reality; I.3.6 [Computer Graphics] Methodology and Techniques – Ergonomics; I.3.6 [Computer Graphics] Methodology and Techniques – Interaction Techniques; I.3.3 [Computer Graphics] Picture/Image Generation – Viewing Algorithms
Additional Keywords: virtual reality, stereoscopic display, head-tracking, image distortion

1 INTRODUCTION

Virtual environments aim to perceptually place the user in an artificial computer-generated world. A key component of creating this illusion is interactive 3D imagery. To generate this imagery, a typical VR system has a location and orientation tracking device, an image generator and one or more displays. The tracking device determines the positions of the user's head and/or eyes and of the displays. The image generator computes the image that each eye would see on a display surface if the eye and the display existed inside the virtual world at their tracked positions. This image is then fed to the physical display. Typical VR systems are configured either as a head-mounted display (HMD) or as a head-tracked display (HTD). In a HMD, the display is attached to a helmet worn by the user, so both the eye points and the display are in continuous motion. In a HTD, the display is stationary, attached to a desk or wall, so only the eye points move. HTD examples are the CAVE [2], the virtual workbench [10] and "fish tank" VR [19].

Most VR systems generate a pair of images, one for each eye. This stereoscopic imagery provides a true 3D image so virtual objects appear to float in front of and behind the physical display surface. Software methods for stereoscopic display are well known [2, 15, 13]. Stereoscopic display for virtual reality has been shown to improve user depth perception and task performance in a variety of tasks [14, 19]. This is not surprising since real world experience shows that stereopsis is an important depth cue especially for objects within the user's personal space (1.5 meters) [3].

Both experience [11] and experimental studies [24, 16] have shown that users with normal stereoscopic vision often have trouble fusing stereo image pairs if the eyes are modeled based on exact eye separation. The common solution is to underestimate the eye separation of the user. This approach solves the image fusion problem but creates a new problem with head-tracked displays.

We observed that underestimating eye separation causes the stereoscopic image to shift and warp with head movement. As the user moves her head forward and back the perceived image will compress and expand. As the user moves her head left and right the perceived image will shift side to side. The images in Figure 1 are indicative of what a user perceives on a stereo HTD with underestimated eye separation. In Figure 1A, a user views a horizontal stereo HTD such as a virtual workbench. The physical display is outlined in black. The gray cube is the model virtual geometry. Figures 1A-1E are a frontal view of this display system. Here the display surface appears as a horizontal black line. While the system internally models a gray cube, the user actually perceives a different shaped object shown in red. Figures 1B and 1C illustrate how the perceived object compresses and expands due to up and down head movement. Figures 1D and 1E illustrate the left/right shifting due to side to side head movement.

This distortion is particularly irksome because the purpose of adding head-tracking to these stereoscopic displays was to remove similar distortions that were observed in earlier *non*-head-tracked systems [9].

It would appear we are stuck between the proverbial rock and a hard place. False eye-separation modeling is necessary to controlling stereo fusion problems and is also instrumental to methods for exaggerating depth of flat scenes [18]. However,

1,2,3 (wartell │ hodges │ ribarsky)@cc.gatech.edu

when applied to stereo HTD's, false eye-separation reintroduces distortions which the tracking component of HTD's was suppose to remove.

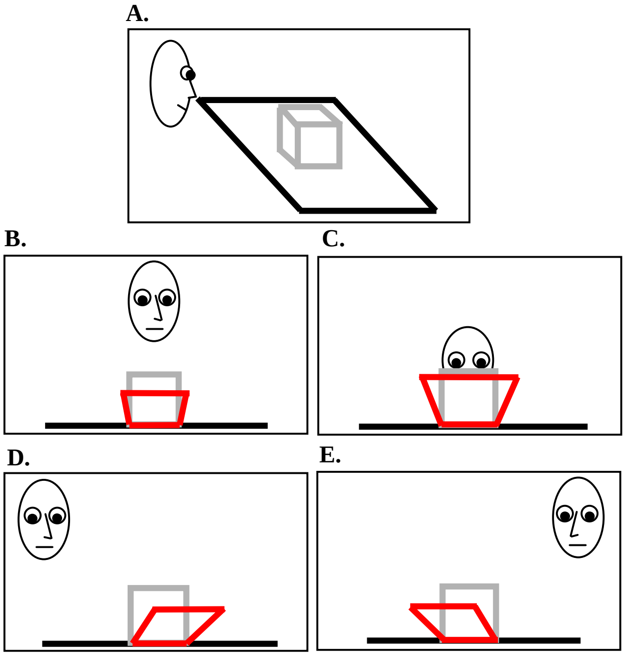

Figure 1: (A) A user viewing a horizontal stereo HTD such as the virtual workbench. The gray cube is a virtual cube. (B-E) are four front views of the user viewing the cube on the stereo HTD. The horizontal black line is the display surface. Underestimated eye separation causes the user to perceive a warped version of the cube shown in red. (A) and (B) illustrate the compression and expansion of the perceived object due to up/down head motion while (C) and (D) illustrate the left/right shifting of the perceived object due to side to side head motion.

This paper presents a novel, highly efficient method for controlling these distortions based on a new analytic description of the distortion. We begin with a geometric construction which describes the distortion and then derive its analytic form, Δ. Δ subsumes previous work on false eye separation distortion and is more general. Analysis of Δ shows that the user will perceive virtual objects to warp and shift when she moves her head even with *perfect* head tracking. We present a predistortion technique for counteracting the shearing component of Δ. Based on this technique, we develop a new method to manage image fusion problems for distant objects and a simple but improved method for enhancing the depth of flat scenes.

2 BACKGROUND AND PREVIOUS WORK

When a user cannot perceive a single 3D image from a stereo image pair, she experiences diplopia (double vision). In a stereoscopic display the occurrence of diplopia is related to various physical attributes of the display system and the geometry of the display environment [9]. The relevant geometric aspects are:

- the distance of the displayed virtual object relative to the display surface
- the eye separation value used in computing the viewing transform
- the distance of the eyes from the display surface

Figure 2 illustrates the situation and some important measurements. The eyes are on the left and a point on a virtual object is on the right. This point is projected onto two points on the projection plane. The screen parallax, *p*, associated with a virtual point is the distance between the projected points. The distance between the eyes and the virtual point also determine the angle, β. Associated with the screen itself is another angle, α Research has shown that if the difference, α-β, is outside a limited range, then diplopia occurs and the 3D depth illusion collapses [24, 9, 15]. This range has a negative limit generally associated with points in front of the projection plane and a positive limit generally associated with points behind the projection plane. The negative limit is called the "crossed-parallax" limit while the positive limit is the "uncrossed-parallax" limit.

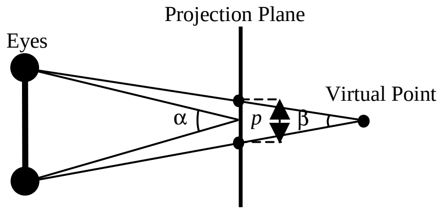

Figure 2: Illustration of the projection of a virtual point onto the projection plane for a user's two eyes. *p* is the horizontal parallax, or distance on the screen between the stereo images of a virtual point. β is the vergence angle of this virtual point. α is the vergence angle of the projection plane itself.

Additional problems with stereoscopic displays are user fatigue and temporary alteration of the visual system's internal coupling of accommodation (eye focus) and convergence (the relative orientation of one eye to the other) [12].

As previously mentioned, the common software technique to minimize these problems in non-head-tracked stereoscopic displays is to model the user's eye separation with a value smaller than the true value. The resulting screen parallaxes and vergence angles are reduced and this minimizes user difficulties. However, when applied to stereo HTD's false eye separation yields the distortions illustrated in Figure 1.

Several researchers find it beneficial to use exaggerated values for the modeled eye separation. Akka [1] reports that users prefer the results of slightly exaggerating the modeled eye separation in a head-tracked stereoscopic display. In Ware [18], the authors dynamically overestimate the modeled eye separation to enhance the perceived depth of terrain. This method was used in a real world application where engineers routed cables along a seabed. Note that this application did not use head-tracking so the stereo distortions introduced by false eye separation modeling would be drowned by the qualitatively similar distortions due to the lack of head tracking.

As discussed in the introduction, these false-eye separation methods induce undesirable distortions in tracked stereo displays. While previous work provides qualitative and quantitative insights into related stereo distortions, none provide a complete description of this distortion.

Researchers [7, 4, 9, 8, 23, 13, 21, 18] investigated various aspects of stereo and monocular distortions. However, none of this previous work describes the stereo distortion due to false eye modeling for stereo HTDs. Ware et al [18] present a brief discussion of the change in the perceived depth of a point for false eye separation modeling in non-headtracked stereo displays. Woods et al [23] derive an analytic description of distortions in stereoscopic tele-operator systems. Woods' treatment assumes the eye axis is parallel to the display plane and that the center of the eyes lies on a line perpendicular to the display and through its center. These assumptions are not true in a stereoscopic HTD system and therefore this previous result does not cover the head-tracked case.

3 DESCRIPTION OF DISTORTION

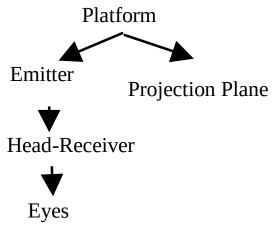

Figure 3: The coordinate system hierarchy for a stereo HTD.

To derive a geometric description of false eye separation distortion, we review and simplify the viewing model used in stereo HTD's. A typical viewing model consists of the coordinate system hierarchy presented in Figure 3. The top coordinate system is the platform coordinate system (PCS). Manipulating this coordinate system moves the user through the virtual space. Directly attached to this coordinate system is the projection plane coordinate system and the emitter coordinate system. The projection plane coordinate system contains the projection plane in its XY plane with the window centered about the origin. The emitter coordinate system simply represents the tracker's emitter. Attached to the emitter coordinate system is the head receiver coordinate system and attached to that is the eye coordinate system. The two eye points are on the x-axis of the eye coordinate system and are symmetric about the origin.

The position and orientation of each child coordinate system relative to its parent are measured physically from the physical display setup along with the view window dimensions. The platform coordinate system's mapping to virtual world coordinates defines the mapping of the physical space of the real world to the virtual space of the virtual world. In addition to specifying the position and orientation, the platform coordinate system can also be uniformly scaled. This causes the virtual world to grow and shrink.

Assuming all the mentioned physical measurements are correct, the virtual eye separation equals the physical separation multiplied by the platform coordinate system's scale. For example, if the modeled eye separation equals the user's true eye separation, say 6 cm, and she views a virtual Earth at a 10^{-6} user scale where the planet appears as a large globe, then the virtual eye separation is 60 km. By our definition this case does not represent overestimated eye separation because the modeled physical eye separation equals the veridical 6 cm.

This paper is *not* concerned with this discrepancy between the virtual eye separation and the physical eye separation. This discrepancy, dependent on PCS scaling, merely scales the virtual world up or down. The world may appear as: a small model, such as the Earth as a globe; a true model, such as a telephone at actual size; or a magnified model, such as an atom at the size of a basketball. This uniform scaling always preserves angles, aspect ratios and parallelism, and maintains the perceived rigidity of the virtual world as the head moves. This paper is concerned with the discrepancy between two further distinguishable values for the physical eye separation: the user's true physical eye separation and the system's modeled physical eye separation. A discrepancy between these values will distort the world by a perspective collineation or homology. The virtual world, at whatever scale it is displayed, will shear and warp with head position and neither angles, aspect ratio nor parallelism will be preserved.

We henceforth ignore PCS scale and the virtual eye separation, and we focus on the modeled and true physical eye separations. The term "eye separation" will now always refer to the physical separations.

Figure 4A illustrates geometrically why false eye separation yields distortions. In Figure 4A, two sets of eye points are illustrated in blue. Within each set the true eye points are on the outside (dark blue) and the modeled eye points are on the inside (light blue). Again the projection plane is the horizontal black line. Below this line, a single modeled point is shown in black along with the perceived point as seen by the left and right eye set positions. For each eye set, the modeled point is projected onto the projection plane through the modeled eyes. These projectors are drawn in black. The true eyes reconstruct a perceived image by finding the intersection of the red lines. These red lines are drawn between an eye and its corresponding projected image point. Note how the perceived point (red) moves as the user moves her head. Also the perceived point is closer to the projection plane than the modeled point. This geometric construction can be applied to a set of points to yield all the distortions illustrated in Figure 1.

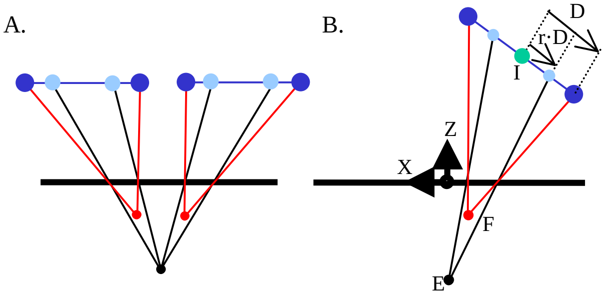

Figure 4: (A) A geometric construction illustrating how false eye modeling distorts the perceived image and how the perceived image moves with head position. (B) A parameterization of geometric construction used to derive the analytic description.

This construction assumes that all the important physical measurements, besides the modeled eye separation, are correct. The construction also assumes any distortion due to curvature of the screen or any optics is negligible or accounted for by other means [4]. Additionally, it assumes that change in the separation of the nodal points of the human eyes during convergence [4] is also negligible or accounted for.

To derive an analytic description of this distortion we parameterized the construction as shown in Figure 4B. First we place the projection plane coordinate system at the center of the projection window with its plane containing the X-Y axes. Next we add a central eye point, *I*. The true left and right eyes (dark blue) are displaced from *I* by the vectors *D* and *–D*. 2|*D*| is the true eye separation. The scalar *r* is the ratio of the modeled eye separation to the true separation. Hence the left and right modeled eyes (light blue) are displaced by *r**D* and *–r**D* respectively, and 2*r*|*D*| is the modeled eye separation. *E* is the modeled point and *F* is the perceived point reconstructed by the user's biological visual system.

The above construction defines a transform, Δ, from an arbitrary point *E* to the point *F*, i.e. *F=Δ(E)*. In [20], we prove that the transformation is the following homology, or projective transform, expressed in projection plane coordinates:

$$\Delta = \begin{bmatrix} 1 & 0 & \dfrac{(1-r)(IxIz + Dx\,Dz\,r)}{Dz^2 r^2 - Iz^2} & 0 \\ 0 & 1 & \dfrac{(1-r)(IyIz + Dy\,Dz\,r)}{Dz^2 r^2 - Iz^2} & 0 \\ 0 & 0 & \dfrac{r(Dz^2 \text{-} Iz^2)}{Dz^2 r^2 - Iz^2} & 0 \\ 0 & 0 & \dfrac{Iz(1-r)}{Dz^2 r^2 - Iz^2} & 1 \end{bmatrix} \qquad (1)$$

In the context of a rendering pipeline the distortion acts as follows. Let a matrix, M_A^B, denote the coordinate transform from coordinate system A to coordinate system B. Let Screen Coordinates be the coordinates after mapping into the canonical parallel-projection view volume, which is an axis-aligned cube with corners (1,1,0) and (-1,-1,-1) [5, p275]. Then the matrix stack during rendering is:

$$M_{Model}^{Screen} = M_{World}^{Screen} \cdot M_{Model}^{World} \qquad (2)$$

Let $[M]_A$ be the representation of a transform M in coordinate system A. Then using false eye separation effectively induces the complete transformation:

$$M'^{Screen}_{Model} = M_{World}^{Screen} \cdot [\Delta]_{World} \cdot M_{Model}^{World} \qquad (3)$$

Therefore, using false eye separation will produce the same perceived 3D image as using the true eye separation and adding $[\Delta]_{World}$ on the viewing stack. Note, since equations (2) and (3) describe virtual space, $[\bullet]_{World}$ will include a scale component inherited from the platform coordinate system. However, when analyzing •, it is convenient to ignore this scale issue and consider the projection plane coordinate system as it exists in the physical world. We can then discuss the effects of • in physical units such as meters and consider how • warps the virtual world at whatever scale it is displayed.

4 PICTORIAL ANALYSIS OF Δ

Figure 1 provided an intuitive understanding of Δ. Figure 5, illustrates Δ more abstractly. The user is represented by her eye points in blue. The true eyes are dark blue while the modeled eye are light blue. The projection plane is the horizontal black line. The model geometry, a mesh, is in black and the perceived geometry, a warped mesh, is red. For underestimated eye separation (*r*=0.5), 6A and 6B show the compression/expansion effect and 6C and 6D show the side to side shifting. Similar distortions occurs with overestimated eye separation [20].

This distortion has many repercussions. A user designing what she perceives to be as a cube may actually have designed a more general truncated pyramid. Equivalent to Wood's [23] observations in teleoperator environments, perceptions of velocity through the environment will also be distorted given this non-linear distortion. Most importantly static, rigid objects will appear to move as the user moves his head. Qualitatively these results are easily verified on real stereoscopic head-tracked displays.

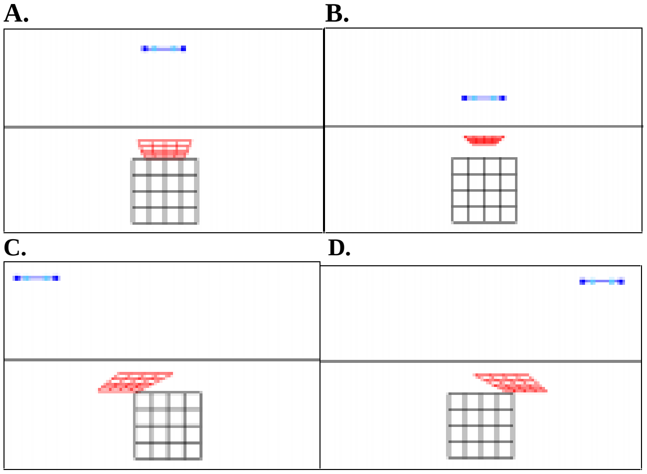

Figure 5: These figures show the that the perceived mesh (red) is a distorted version of the modeled mesh (black). The eye points are in blue and the projection plane is the horizontal black line. For underestimate eye separation (r=0.5), A and B illustrate the compression and expansion while C and D illustrate the side to side shifting.

5 QUANTITATIVE ANALYSIS OF Δ

For a quantitative analysis of Δ, we assume neither the modeled eye points nor the true eye points are embedded in the projection plane. These cases lead to degenerate, singular mappings in both Δ and the original construction [20]. Since these embedded cases are rare, ignoring them is permissible.

5.1 Maximum Depth Plane

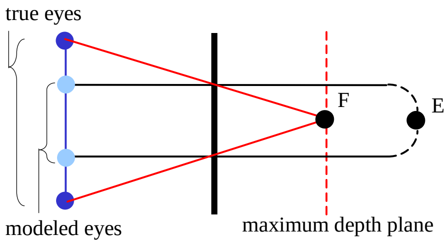

Figure 6: Illustration of the maximum depth plane of perceived space due to underestimating eye separation

We can use Δ^{-1} to compute the maximum possible depth in perceived space when the modeled eye separation is smaller than the true eye separation (*r* < 1). The existence of a maximum depth in the perceived space has been noted before [23, 18]. Figure 6 illustrates this idea. For a point beyond the projection plane the screen parallax reaches its maximum value, equal to the modeled eye separation, for a point infinitely far away, *E*. This places a limit on the depth of the reconstructed perceived points, *F*.

For a non-degenerate viewing configuration, Δ is non-singular and hence Δ^{-1} exists. Like Δ, Δ^{-1} is a homology so it has a plane, *P*, of ordinary points which are mapped to ideal points (points at infinite). This plane is called the vanishing plane since these points have no image in Euclidean space. Δ being the inverse of Δ^{-1} maps these ideal points back to the affine plane *P*. These ideal points represent the points lying infinitely far beyond the projection plane that get mapped to the maximum depth plane. *P* then is precisely this maximum depth plane. The equation for the maximum depth plane is the vanishing plane of Δ^{-1}. It is easy to find the vanishing plane of a perspective matrix [6]. With this insight the maximum depth plane is:

$$z = \frac{r\left(Dz^2 - Iz^2\right)}{Iz\left(1 - r\right)} \tag{4}$$

Equation (4) illustrates how the maximum depth plane position varies with the head position's z-component. This helps explain the head-position dependent squashing of perceived space illustrated in Figure 7. Here the perceived grid compresses as the head moves towards the projection plane. This motion also brings the maximum depth plane (the dash red line) closer in.

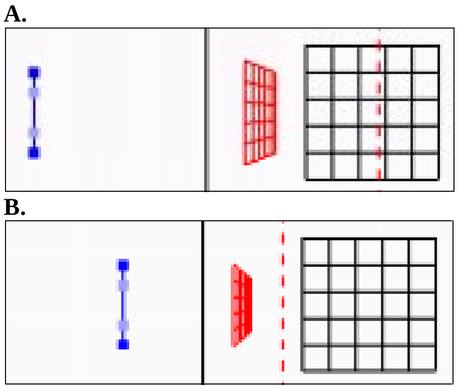

Figure 7: Perceived grid (red) squashed towards view plane. Note maximum depth plane (dashed red line). Again the true eyes are dark blue and the modeled eyes are light blue.

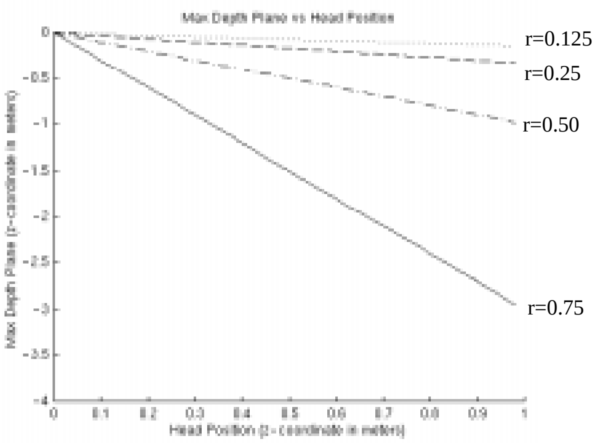

Figure 8: Plot of the position of the maximum depth plane versus user head position for various modeled-to-true eye separation ratios (*r*).

Figure 8 plots the position of the maximum depth plane as a function of viewer head position (Iz) for several eye separations ratios (*r*): 0.75 (solid), 0.5 (dash-dot), 0.25 (dash) and 0.125 (dot). Note, Figure 8 assumes the head is parallel to the projection plane (*dz*=0); however, even for non-parallel case *dz* is typically small compared to *Iz*. In Figure 8, the maximum depth plane position is linear with respect to the head position while it varies non-linearly with *r*. Smaller modeled eye separations produce a closer maximum depth plane and hence a greater compression of the perceived space.

5.2 Side to Side Shifting

Previously figures 1C and 1D illustrated the sideways shifting induced by false eye-separation. Here we examine this shifting more rigorously. We plot the x-coordinate difference of a modeled point, *E*, from its distorted point, *F*, as a function of head position. For simplicity, assume the eyes are parallel to the projection plane and are contained in the X-Z plane (dz,dy=0). Fix the central eye's (I) z-coordinate to 1 meter and then vary the central eye's x-coordinate so that the head moves side to side. In this case, *Fx* and hence *Fx-Ex,* varies linearly with *Ix*.

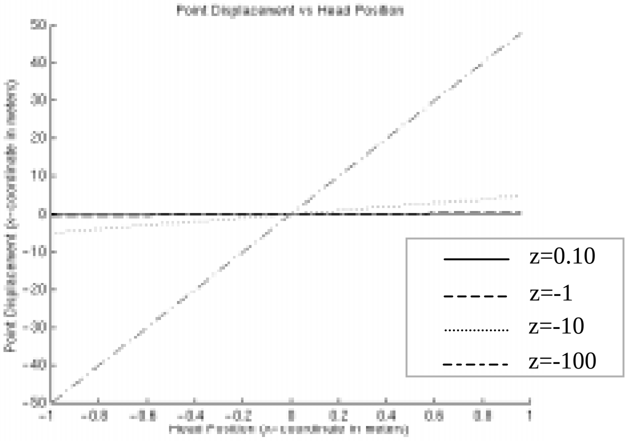

Figure 9: Plot of the displacement of a perceived point from its modeled location versus head position. Head position, *Ix*, varies from –1 to 1; *r* is 0.5; eye-separation is 0.065. Plots are drawn for a model point at various z coordinates.

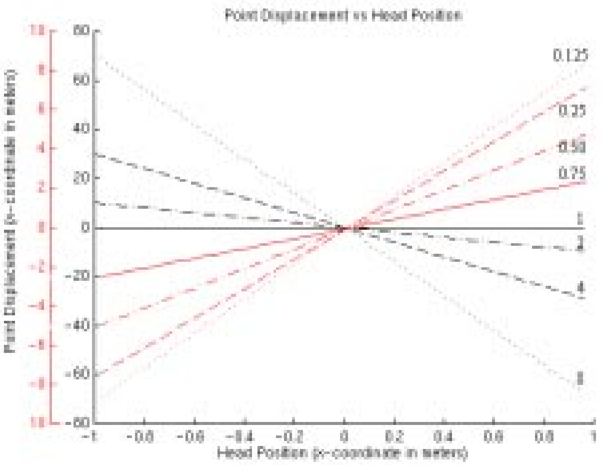

Figure 10: Plot of the displacement of a perceived point from its modeled position (Fx-Ex) versus head position. The modeled point is at (0,0,-10). *r*, the eye separation ratio, is varied over a range of values: 0.125 to 0.75 in red and 1 to 8 in black.

In Figure 9, *Fx-Ex* is plotted against *Ix*. *Ix* varies from –1 to 1; *r* is 0.5; eye-separation is 0.065m. Plots are drawn for a model point a *Ez*=0.10 (solid),*Ez*=-1 (dashed), *Ez*=-10 (dotted) and *Ez*=-100 (dash-dot). Sensitivity to head position grows with object depth, with *Ez*=0.10m ranging up to 0.05m and *Ez*=-100 m ranging up to –50 m.

Figure 10 shows the effect of different values for *r* for a model point at (0,0,-10). In Figure 10 in red, r is 0.75 (solid), 0.5 (dash-dot), 0.25 (dash) and 0.125 (dot). In Figure 10 in black, r is 1 (solid), 2 (dash-dot), 4 (dash) and 8 (dot). Generally, as we move away from using true eye separation, *r*=1, the shifting grows more sensitive to head movement. Note also the change from positive to negative slope as *r* goes from less to greater than one. This represents a reversal in the direction of the shifting.

This discussion illustrates the behavior of the distortions shifting. The plots show the shift grows quite large especially for modeled eye-separations far from the true value (*r*=1).

6 REMOVING ARTIFACTS OF Δ

To remove some of the artifacts of false eye modeling while maintaining the effect on screen parallax, we can derive a predistortion transform *Q* to place on the matrix stack:

$$M^{Screen}_{World} \cdot \left[Q\right]_{World} \cdot M^{World}_{Model} \tag{5}$$

With false eye separation, this results in an effective matrix stack:

$$M'^{Screen}_{Model} = M^{Screen}_{World} \cdot \left[\Delta \cdot Q\right]_{World} \cdot M^{World}_{Model} \tag{6}$$

Q should cancel the undesirable aspects of Δ while retaining the effect on screen parallax and perceived depth.

Q = Δ^{-1} is not useful since it cancels all the effects of false eye modeling including the desired ones. The discussion of the maximum depth plane in 5.1, illustrates in that the changes to perceived depth due to false eye separation are inherently perspective in nature. Therefore, the perspective aspect of Δ is not removable. The side to side shifting effect, however, can be removed.

6. 1 Δ_{Shear}^{-1} Predistortion

Predistorting the world by the inverse shear component of Δ will remove the sideways shifting. To extract this component, Δ is first decomposed into a shear of X and Y along Z, Δ_{shear}; a Z scale, Δ_{scale}; and a pure projection, $\Delta_{Project}$ [20]. From the detailed decomposition in [20], we can find the inverse of Δ_{shear}:

$$\Delta^{-1}_{Shear} = \begin{bmatrix} 1 & 0 & \dfrac{-\left(1-r\right)\left(IxIz + Dx\,Dz\,r\right)}{Dz^2 r^2 - Iz^2} & 0 \\ 0 & 1 & \dfrac{-\left(1-r\right)\left(IyIz + Dy\,Dz\,r\right)}{Dz^2 r^2 - Iz^2} & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{7}$$

The matrix stack we should build at run-time is:

$$M^{Screen}_{World} \cdot \left[\Delta^{-1}_{Shear}\right]_{World} \cdot M^{World}_{Model} \tag{8}$$

When using false eye separation, predistorting world space with $\Delta_{Shear}{}^{-1}$ cancel the shear component of Δ, yielding the complete effective transform:

$$M^{Screen}_{World} \cdot \left[\Delta_{Project} \cdot \Delta_{Scale}\right]_{World} \cdot M^{World}_{Model} \tag{9}$$

This technique removes the most objectionable part of Δ, the head dependent shearing. Figure 11 illustrates this. Figures 11A and 11E show the original grid (black) and the perceived grid (red) without predistortion. Next 11B and 11F show the perceived mesh after the original mesh is predistorted into a new mesh (blue) by $\Delta_{Shear}{}^{-1}$. With the predistortion the perceived mesh is stationary even as the eye moves from left to right (11B to 11F).

6.2 α-Predistortion

Experimentally, we find it useful to use a slightly altered version of the $\Delta_{Shear}{}^{-1}$ technique. If the central eye, *I*, lies on a special curve, $\Delta_{Shear}{}^{-1}$ is the identity. It is useful to position this curve so that the default viewing position for a given stereo head-tracked display lies on the curve. This locks the perceived objects in place as seen from this standard view position. For the typical viewing position where an eye axis is parallel to the view plane, this curve is a line perpendicular to the display surface.

With a vertical display system, the fixed line could be centered horizontally on the display and then positioned vertically to coincide with the average user's eye level. For a horizontal display, such as the virtual workbench, the fixed line could be centered horizontally on the display and then translated forward perhaps a meter in front of the workbench.

To shift this fixed line we derive the following predistortion matrix,*α*, to replace $\Delta_{Shear}{}^{-1}$:

$$\alpha = \begin{bmatrix} 1 & 0 & \dfrac{-\left(1-r\right)\left(IxIz + Dx\,Dz\,r - Iz\,Fx\right)}{Dz^2 r^2 - Iz^2} & 0 \\ 0 & 1 & \dfrac{-\left(1-r\right)\left(IyIz + Dy\,Dz\,r - Iz\,Fy\right)}{Dz^2 r^2 - Iz^2} & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{10}$$

where

Fx and Fy are the x and y coordinates of the fixed line

Figure 11 illustrates the use of the predistortion *α* with different fixed lines (dash gray). The top row illustrates eyes on the left while the bottom row illustrates eyes on the right. In the first column, *α* is not used. In the next 3 columns *α* is used with the fixed line in at center, on the left, and on the right. Using *α* with the fixed line on the left, the perceived grid for the left head position *is not altered* by *α* (compare C to A), while the perceived grid for the right head position *is altered* by *α* (compare E to G). On the other hand, using *α* with a fixed line on the right, the perceived grid for the left head position *is altered* by *α* (compare D to A), while the perceived grid for the right head position *is not altered* by *α* (compare H to D). Hence changing the position of the fixed line determines which viewpoints are locked in place. This fixed line can be positioned to contain the typical viewing position in a given head-tracked stereo display system.

In OpenGL [22], *α*-predistortion can be implemented as follows. At every frame compute *α*. Next compute *[α]*$_{view}$. *[α]*$_{view}$ is *α* relative to the OpenGL view coordinate system.

Recall OpenGL combines the affine model and view transforms on a single “ModelView” matrix stack while the projective frustum transform goes on the “Projection” matrix stack. Typically, the first transform placed on the ModelView stack is that which maps world coordinates to view coordinates. This transform consists of a rotation and a translation accounting for the location of the eye point and the orientation of the view plane. Since the orientation of the view coordinate system and the projection plane coordinate system are the same, α is mapped from projection plane coordinates to view coordinates as follows: $[\alpha]_{view} = S^{-1} \bullet T^{-1} \bullet \alpha \bullet T \bullet S$ where T is a translation by the coordinates of the eye in projection plane coordinates and S is the platform scale. $[\alpha]_{view}$ must be placed between the last transform on the Projection stack and the first transform on the ModelView stack. To avoid affecting lighting, place $[\alpha]_{view}$ on the Projection matrix stack as the last transform. Also note that for applications with application level bounding-box to view-volume culling, the α transform must be taken into account.

7 APPLICATIONS OF α-PREDISTORTION

False-eye separation modeling is used either to avoid diplopia by underestimating the eye separation or to enhance stereoscopic depth by exaggerating eye separation. For head-tracked displays, this induces a distortion Δ. While we tested and verified that α-enhanced false-eye modeling removes the shearing for both underestimated and overestimated eye separation, we argue that exaggerated eye separation, even with α-predistortion, is not useful.

With exaggerated stereo two choices exist:

- produce a distorted image that shears with head-movement and does not preserve parallelism

or

- produce a distorted image that does not shear but still does not preserve parallelism (using α-predistortion)

In the best case exaggerated stereo still distorts the image by a homology about the projection plane. But why use a depth enhancing distortion that will necessarily map cubes to truncated frustums (i.e. not-preserve parallelism)? We are better off simply scaling the world perpendicular the projection plane in order to enhance the depth.

However, underestimating eye separation to minimize diplopia is still useful. Because Δ is a homology it has the effect of bringing points at infinity to some fixed plane beyond the projection plane. So by underestimating user eye separation we can map the *entirety* of space beyond the projection plane to a finite region between the projection plane and some maximum depth plane (5.2). (No affine transform can do this). Now we can set this maximum depth plane to the maximum fusible depth plane, the plane that delimits the farthest points that typical users can fuse. Rewriting Southard’s [15] equation (6) for the maximum fusible depth plane using our nomenclature:

$$far\_fusible = Iz - \frac{2\left|D\right|Iz}{2\left|D\right| - Iz\theta_{\max}} \tag{11}$$

$$\text{where } \theta_{\max} \text{ is the positive uncrossed vergence angle limit}$$

Now from equation (4) we can solve for the eye separation ratio, r, that will bring all points infinitely beyond the view plane into the fusible region delimited by *far_fusible*:

$$r = \begin{cases} \dfrac{far\_fusible \cdot Iz}{Dz^2 - Iz^2 + far\_fusible \cdot Iz}, & \text{if } 2\left|\mathrm{D}\right| - Iz\theta_{\max} > 0 \\ 1, & \text{otherwise} \end{cases} \tag{12}$$

The r=1 case occurs when the user is far enough from the projection plane so that the maximum depth plane is at infinity, i.e. all far space is fusible. In this case we use the true eye separation (r=1). Finally, applying α-predistortion removes the left/right shearing.

The first virtue of this technique over previous methods is that it does not throw out geometry that the user cannot fuse. Previous methods set the far clipping plane to *far_fusible* [15]. In contrast, the new method throws out nothing; instead it maps the *entire* geometry data set into the fusible depth range. Additionally compared to [18], the technique does not require computing scene depth at every frame and it does not have the left/right shearing.

8 CONCLUSIONS

We have presented a novel analytic description of the distortion induced by false eye-separation modeling for a head at an arbitrary position and orientation. We analyzed the effects of this distortion as it relates to head-tracked stereoscopic displays. We then presented the α-predistortion technique for counteracting the removable artifacts of this distortion. Finally we described new methods for controlling uncrossed diplopia and enhancing depth.

9 FUTURE WORK

An analytic description of the distortion due to false-eye separation can aid studies of the effects of a change in eye-separation due to convergence [4]. Second, further interface development, exploration and user study is warranted for α-predistortion and for the techniques presented for depth enhancement and diplopia control. Issues are:

- What value to use for $\theta_{\max}$ for a variety of display technologies? (Yeh and Silverstein’s 1.57 is very conservative and is based on color CRT’s with a particular phosphor set).
- Extensions for managing crossed-parallax diplopia.
- How these techniques interact with the wide variety of HTD applications?
- How these techniques affect user performance in basic and more applied tasks in HTD’s?

Acknowledgements

This work was performed in part under contracts N00014-97-1-0882 and N00014-97-1-0357 from the Office of Naval Research. Support was also provided under contract DAKF11-91-D-004-0034 from the U.S. Army Research Laboratory.

References

[1] Robert Akka. Utilizing 6D head-tracking data for stereoscopic computer graphics perspective transformations. *Proceedings of the SPIE - The International Society for Optical Engineering, Stereoscopic Displays and Applications IV*, 1915: 147-154, Feb. 1993.

[2] C. Cruz-Neira, D.J. Sandin, , T.A. DeFanti, Surround-screen projection-based virtual reality: the design and implementation of the CAVE. In *SIGGRAPH 93 Conference Proceedings*, Annual Conference Series, pages 135-42. ACM SIGGRAPH, Addison Wesley, August 1993.

[3] James E Cutting. How the eye measures reality and virtual reality. *Behavioral Research Methods, Instruments & Computers*, 29(1): 27-36, February 1997.

[4] Michael Deering. High Resolution Virtual Reality. In *Computer Graphics (SIGGRAPH 92 Conference Proceedings)*, volume 26, pages 195-202. Addison Wesley, July 1992.

[5] James D. Foley, Andries Van Dam, Steven K. Feiner, John F. Huges. *Computer Graphics: Principles and Practice*. Addison-Wesley Publishing Company. 1992.

[6] Ronald N Goldman. Decomposing Projective Transformations. In David Kirk, editor, *Computer Graphics Gems III*. Boston : Harcourt Brace Jovanovich, 1992.

[7] Larry F. Hodges, David F. McAllister. Rotation algorithm artifacts in stereoscopic images. *Optical Engineering*. 29(8): 973-976, August 1990.

[8] Larry F. Hodges, Elizabeth Thorpe Davis. Geometric Considerations for Stereoscopic Virtual Environments. *Presence*, 2(1)*:* 34-42, Winter 1993.

[9] Larry F. Hodges. Tutorial: Time-Multiplexed Stereoscopic Computer Graphics. *IEEE Computer Graphics and Applications*. 12(2): 20 –30, March 1992.

[10] W. Krüger, B. Fröhlich. The Responsive Workbench (virtual work environment). *IEEE Computer Graphics and Applications*. 14(3):12-15. May 1994.

[11] Lenny Lipton. *Foundations of the Stereoscopic Cinema: A Study in Depth*. Van Nostrand Reinhold, 1982.

[12] M. Mon-Williams, J.P. Wann, S. Rushton. Design factors in stereoscopic virtual-reality displays. *Journal of SID*, 3/4: 207-210. 1995.

[13] Warren Robinett, Richard Holloway. The Visual Display Transformation for Virtual Reality. *Presence*, 4(1): 1-23. Winter 1995.

[14] Louis B. Rosenberg. The Effect of Interocular Distance upon Operator Performance using Stereoscopic Displays to Perform Virtual Depth Tasks. In Proceedings of *IEEE Virtual Reality Annual International Symposium 93*, 27-32. Sept. 1993.

[15] David A. Southard. Viewing model for virtual environment displays. *Journal of Electronic Imaging*, 4(4): 413–420. October 1995.

[16] R. Troy Surdick, Elizabeth T. Davis, Robert A. King, Larry F. Hodges. The Perception of Distance in Simulated Displays. *Presence*, 6(5): 513-531, October 1997.

[17] John P. Wann, Simon Rushton, Mark Mon-Williams. Natural Problems for Stereoscopic Depth Perception in Virtual Environment. *Vision Research*. 35(19): 2731-2736, 1995.

[18] Colin Ware, Cyril Gobrecht and Mark Paton. Algorithm for dynamic disparity Adjustment. In *Proceedings of the SPIE - The International Society for Optical Engineering*. *Stereoscopic Displays and Virtual Reality Systems II*, 2409:150-6, Feb. 1995.

[19] Colin Ware, Kevin Arthur and Kellogg S. Booth. Fish Tank Virtual Reality. In proceedings of *InterChi '93*, pages 37-41. April 1993.

[20] Zachary Wartell, Larry F. Hodges, William Ribarsky. The Analytic Distortion Induced by False-Eye Separation in Head-Tracked Stereoscopic Displays. Georgia Institute of Technology, GVU Technical Report, no. 99-01, 1999. (See also *Computer Graphics Proceedings CD-ROM SIGGRAPH 99*).

[21] B.A. Watson, L. F. Hodges. Using texture maps to correct for optical distortion in head-mounted displays. In proceedings of *IEEE Virtual Reality Annual International Symposium 95* (VRAIS '95), pages 172-178.

[22] Mason Woo, Jackie Neider, Tom Davis. *OpenGL Program Guide*. Addison-Wesley Developers Press. Reading, Massachusetts. 1997.

[23] Andrew Woods, Tom Docherty, Rolf Koch. Image Distortion in Stereoscopic Video Systems. In *Proceedings of the SPIE – The International Society for Optical Engineering, Stereoscopic Displays and Applications IV*, 1915: 36 – 48, 1993.

[24] Yei-Yu Yeh and Louis .D. Silverstein. Limits of Fusion and Depth Judgements in Stereoscopic Color Displays. *Human Factors*, 32(1): 45-60, Feb. 1990.

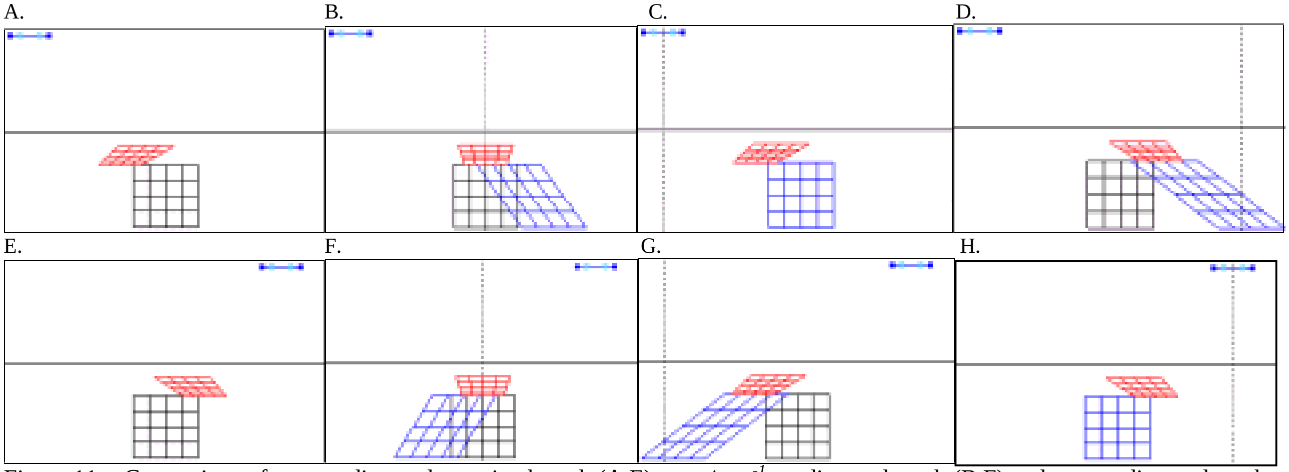

Figure 11: Comparison of non-predistorted perceived mesh (A,E) to a Δ_{shear}^{-1}-predistorted mesh (B,F) and a α-predistorted mesh with the fixed line on the left (C,G) and the fixed lined on the right (D,H). The fixed line is the dashed, gray vertical line. The true eyes are dark blue and the modeled eyes are light blue. The perceived mesh is red. The modeled mesh is black and the predistorted mesh is blue. Comparing (A) and (E) to their corresponding rows, note that both predistortion methods remove the sideways shifting of the perceived mesh. Using α-predistortion (C,D,G,H) allows relocation of the fixed line which represents the set of viewpoints whose views are unaffected by α-predistortion.