A Geometric Comparison of Algorithms for Fusion Control in Stereoscopic HTDs: Supplementary Derivations

Technical Report GVU-GIT-01-08b

7/5/01

Zachary Wartell, Larry F. Hodges, William Ribarsky

***Notice**: This document is a supplement for "A Geometric Comparison of Algorithms for Fusion Control in Stereoscopic HTDs" GVU Tech Report, GIT-GVU-01-08a by the same authors (see Reference [2]).*

PART A: IMAGE SCALING DISTORTION

A1 Parallel Case

The following figure illustrates the distortion induced by image scaling for a head at an arbitrary position but oriented parallel to the projection plane. The eye points are on the left in blue and the projection plane is in the X-Y plane. The projection window is centered about the origin. (Note, Figure A-1 only shows a portion of the projection window so the window does not actually appear centered in the diagram). **E** is the modeled object point and **F** (red) is the equivalent displayed object point. The user's central eye point is at **I**. The left eye, **D**, is displaced by **d** from **I** and the right eye, **A**, is displaced by **-d**. 2|**d**| is the true eye separation. We assume correct modeling of the eye separation. **E** is projected onto the points **H** and **G** on the projection plane. Image scaling by scalar factor *s* scales points **H** and **G** into points *s*·**H** and *s*·**G**. These scaled points are those the user sees. This image manipulation has the same effect as if the 3D point **E** were mapped to **F**. Note for this parallel case **A**,**D**,**d** and **E** are <u>not</u> restricted to be in the XZ plane.

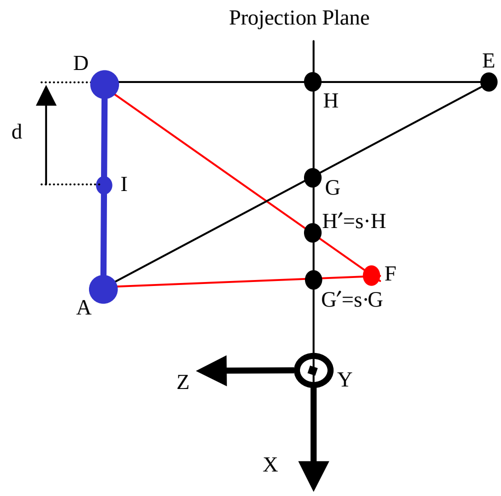

Figure A-1: This is an abstract diagram of a user viewing a stereoscopic HTD at a parallel orientation. See above text for details.

A1.1 From Figure A-1:

$\mathbf{A} = \mathbf{I} - \mathbf{d}$
$\mathbf{D} = \mathbf{I} + \mathbf{d}$
$d_z = 0$

A1.2 Solve for **H**:

Equation of line **DH** is:

$$\mathbf{P} = (\mathbf{E}-\mathbf{D})t + \mathbf{D}$$

At $z = 0$:

$$t = \frac{D_z}{D_z - E_z}$$

So:

$$\mathbf{H} = (\mathbf{E} - \mathbf{D})\frac{D_z}{D_z - E_z} + \mathbf{D}$$

Or from A1.1:

$$\mathbf{H} = (\mathbf{E} - \mathbf{I} - \mathbf{d})\frac{I_z + d_z}{I_z + d_z - E_z} + \mathbf{I} + \mathbf{d}$$

A1.3 Solve for **G**:

Using arguments similar to A1.2:

$$\mathbf{G} = (\mathbf{E} - \mathbf{A})\frac{A_z}{A_z - E_z} + \mathbf{A} = (\mathbf{E} - \mathbf{I} + \mathbf{d})\frac{I_z - d_z}{I_z - d_z - E_z} + \mathbf{I} - \mathbf{d}$$

A1.4 Solve for F_x:

To begin:

$$\mathbf{F} = \overline{\mathbf{A}s\mathbf{G}} \cap \overline{\mathbf{D}s\mathbf{H}}$$

In [1] we derived a similar result for distortions due to false eye separation. By substituting $s{\cdot}\mathbf{G}$ for the **G** and $s{\cdot}\mathbf{H}$ for the **H** for in the final result of Appendix 1.4 in [1] (bottom of page 13) we have

$$F_x = \frac{A_z D_x\, sG_x - A_x D_z\, sH_x + (D_z - A_z)\, sG_x\, sH_x}{A_z D_x - A_x D_z + D_z\, sG_x - A_z\, sH_x}$$

Our previous manual derivation for this type of expression was quite laborious and tedious [1], therefore we use a commercial software algebra solver. To find F_x, we use Mathematica [3] with the following input file:

```
Hx=(Ex]]]([(]]]]
]=(Ex]]]([(]]]]

] = ] ]]
]= ]]]
] = ] ]]
]= ]]]
]= ]

(]]]
]Ex]] ]] ]] ]]Hx ](]] ]Hx]
]= ([](]]
]= ]

]= Ex]] ]] ]]]]] ]]]Hx]
]= ([](]]
]= ]

] = ]
] = ]]]x]]x]]
```

The final result is:

$$F_x = \frac{E_x I_z s}{E_z(s-1) + I_z}$$

A1.5 Solve for F_y**:**

Solving for F_y uses a parallel derivation to F_x, yielding:

$$F_y = \frac{E_y I_z s}{E_z(s-1)+I_z}$$

A1.6 Solve for F_z:

Using the initial results from Appendix 1.6 in [1] (bottom of page 23) and substituting *s* **H** for **H** and *s* **G** for **G**:

$$F_z = \frac{A_z D_z s H_x - A_z D_z s G_x}{-A_z D_x + A_x D_z - D_z s G_x + A_z s H_x}$$

Expression for **H** and **G** are in the above sections A1.2 and A1.3. To find F_z, we use Mathematica [3] with the following input file:

```
Hx=(Exxx[[Exx
x=(Exxx[[Exx

x = x ]x
]= ]]]
x = x ]x
]= ]]]
]= ]

(]]]
]= Ex]]]Hx ]]]]x]
]= [E[E]
]= ]

]= Ex]x ]x ]]]]x ]]]Hx]
]= [E[E]
]= ]

]= ]
]= EE]
```

This yields:

$$F_z = \frac{E_z I_z s}{E_z(s-1)+I_z}$$

A1.7 Rewrite in Matrix Form:

Combining sections A1.4-A1.6 and using column vector notation the distortion matrix is:

$$\boldsymbol{\Delta}_{sc}^{p} = \begin{pmatrix} I_z s & 0 & 0 & 0 \\ 0 & I_z s & 0 & 0 \\ 0 & 0 & I_z s & 0 \\ 0 & 0 & s-1 & I_z \end{pmatrix} = \begin{pmatrix} s & 0 & 0 & 0 \\ 0 & s & 0 & 0 \\ 0 & 0 & s & 0 \\ 0 & 0 & \frac{s-1}{I_z} & 1 \end{pmatrix}$$

The last simplification is possible since homology matrices are only unique up to a scale factor and we assume the eye axis center is not embedded in the projection plane.

A2 Non-Parallel Case

The following figure illustrates the distortion induced by image scaling for a head at an arbitrary position and orientation. The diagram uses the same coloring and naming conventions as Figure A-1. **E** is the modeled object point and **F** is the equivalent displayed object point. The user's central eye point is at **I**. **d** is the vector to the left eye. **E** is projected onto the points **H** and **G**. Image scaling by scalar factor *s* scales points **H** and **G** into points **H′**=*s*·**H** and **G′**=*s*·**G**. As discussed elsewhere [1], lines **DH′** and **AG′** do not generally intersect. Therefore, we restrict ourselves to a simple, special case where they do intersect. **A**,**D**,**d** and **E** are restricted to be in the XZ plane as shown in Figure A-2. Since A_y,D_y,d_y,E_y equal zero and the y coordinates of all dependent points, **H**, **G**, etc., are also zero, this yields a well-defined point at location **F**.

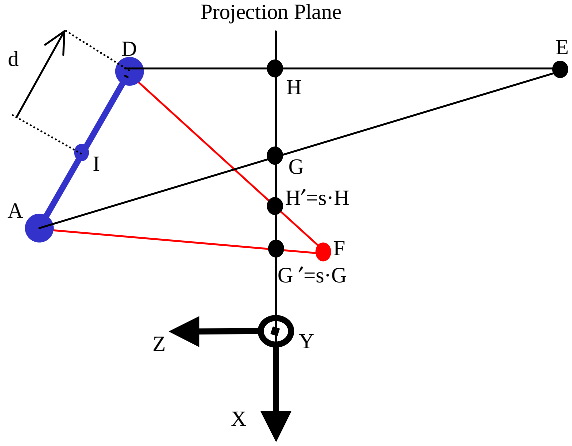

Figure A-2: This is an abstract diagram of a user viewing a stereoscopic HTD at an arbitrary orientation. See above text for details.

The derivation is parallel to section A1. When computing the expressions for F_x and F_z just remove the line 'dz=0' in the respective Mathematica files (section A1.4 and A1.6). F_y is simply 0. These alterations account for the fact that we are limiting ourselves to the XZ plane and that the eye axis is no longer parallel to the screen. The resulting expression is:

$$\Delta_{sc}^{XZ} : F_x = \frac{\left(E_x (d_z I_x - d_x I_z)(d_z^2 - I_z^2) s + E_x^2 \left(d_z s^2 \left(-d_z^2 + I_z^2 \right) \right) + E_x E_z s\left(-2 d_z I_x I_z (-1+s) + d_x \left(-d_z^2 - I_z^2 + 2 d_z^2 s \right) \right) - E_z^2 (d_z)(d_x^2 - I_x^2)(-1+s) s \right)}{w}$$

$$F_z = \left(E_x E_z \left(-d_z s\left(d_z^2 - I_z^2 \right) \right) + E_z \left(d_z^2 - I_z^2 \right) \left((d_z I_x - d_x I_z) s + E_z^2 \left(d_x s \left(d_z^2 - I_z^2 \right) \right) \right) \right) / w$$

$$w = \left(E_x \left(d_z s \left(-d_z^2 + I_z^2 \right) \right) + E_z \left(d_x I_z^2 (-2+s) - 2 d_z I_x I_z (-1+s) + d_x d_z^2 s \right) + E_z^2 (d_z I_x - d_x I_z)(-1+s) + (d_z I_x - d_x I_z)(d_z^2 - I_z^2) \right)$$

PART B: IMAGE SHIFTING DISTORTION

B1 Parallel Case

The following figure illustrates the distortion induced by image shifting for a head at an arbitrary position but oriented parallel to the projection plane. The labeling and color scheme are the same as in Figure A-1. The user's central eye point is at **I**. **d** is the vector to the left eye. **E** is projected onto the points **H** and **G** on the projection plane. Window translates points **H** and **G** by distance τ along vector **T** which is the XY-projected unit vector of **d**. This yields **H′=H-** τ**T** and **G′=G+** τ**T** which are seen by the user.

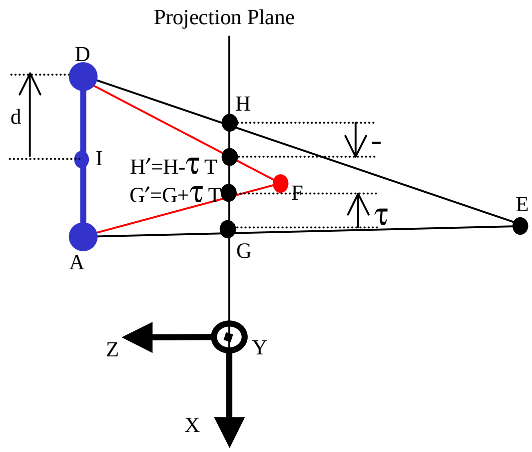

Figure B-1: This is an abstract diagram of a user viewing a stereoscopic HTD a parallel orientation. See above text for details.

B1.1 From Figure B-1:

$\mathbf{A} = \mathbf{I} - \mathbf{d}$
$\mathbf{D} = \mathbf{I} + \mathbf{d}$
$d_z=0$
$\mathbf{T} = (d_x/\mathrm{sqrt}(d_x^2+ d_y^2),\ d_y/\mathrm{sqrt}(d_x^2+ d_y^2), 0)$

B1.2 Solve for F_x:

To begin:

$$\mathbf{F} = \overline{\mathbf{AG}'} \cap \overline{\mathbf{DH}'}$$

where

$$\mathbf{G}' = \mathbf{G} + \tau\mathbf{T}$$
$$\mathbf{H}' = \mathbf{H} - \tau\mathbf{T}$$

Expressions for **H** and **G** the same as in A1.2 and A1.3. In [1] we derived a similar result for distortions due to false eye separation modeling. By substituting **G′** for the **G** and **H′** for the **H** in the final result of Appendix 1.4 in [1] (bottom of page 13) we have:

$$F_x = \frac{A_z D_x\ G'_x - A_x D_z\ H'_x + (D_z - A_z)\ G'_x\ H'_x}{A_z D_x - A_x D_z + D_z\ G'_x - A_z\ H'_x}$$

Manual derivations for this type of expression are quite tedious [1], therefore we use Mathematica [3] with the following input file:

```
]

Hx=(Ex]]([]]
]=(Ex]]([]]

] = ] ]]] ]]

Hx = Hx ]]]
] = ] ]]]

] = ] ]]
]= ]]]
] = ] ]]
]= ]]]
]= ]

(]]]
]Ex]] ] ]] ]Hx ](]] Hx]
]= ([([]
]= ]

]= Ex]] ]] ]]] ] ]]Hx]
]= ([([]
]= ]

] = ]
] = ][Ex[][Ex[]
```

The final result is:

$$F_x = \frac{E_x\, I_z \sqrt{d_x^{\,2} + d_y^{\,2}} - E_z\, I_x \tau + I_x I_z \tau}{-E_z \tau + I_z \left(\tau + \sqrt{d_x^{\,2} + d_y^{\,2}}\right)}$$

B1.3 Solve for F_y:

Solving for F_y uses a parallel derivation to F_x (B1.2), yielding:

$$F_y = \frac{E_y\, I_z \sqrt{d_x^{\,2} + d_y^{\,2}} - E_z\, I_y \tau + I_y I_z \tau}{-E_z \tau + I_z \left(\tau + \sqrt{d_x^{\,2} + d_y^{\,2}}\right)}$$

B1.4 Solve for F_z:

Substituting **H′** for **H** and **G′** for **G** in results from Appendix 1.6 in [1] (bottom of page 23):

$$F_z = \frac{A_z D_z H'_x - A_z D_z G'_x}{-A_z D_x + A_x D_z - D_z G'_x + A_z H'_x}$$

Next we use Mathematica with the following input file:

]

Hx=(Ex]]]]]]]]
]=(Ex]]]]]]]

] =]]]]]]

Hx = Hx]]]
] =]]]]

] =]]]
]=]]]
] =]]]
]=]]]
]=]

(]]]
]= Ex]] Hx]]]]]
]=]]]]]
]=]

]= Ex]]]]]]]]]] Hx]
]=]]]]]
]=]

]=]
]=]]]

This yields:

$$F_z = \frac{E_z I_z\left(\sqrt{d_x^2 + d_y^2} - \tau\right) + I_z^2 \tau}{-E_z \tau + I_z\left(\tau + \sqrt{d_x^2 + d_y^2}\right)}$$

B1.5 Rewrite in Matrix Form:

Combining Sections B1.2-B1.4 and using a column vector notation yields the matrix:

$$\boldsymbol{\Delta_{sh}^{p}} = \begin{bmatrix} I_z Q & 0 & -I_x \tau & I_x I_z \tau \\ 0 & I_z Q & -I_y \tau & I_y I_z \tau \\ 0 & 0 & I_z Q - I_z \tau & I_z^2 \tau \\ 0 & 0 & -\tau & I_z(\tau + Q) \end{bmatrix} = \begin{bmatrix} Q & 0 & -\frac{I_x}{I_z}\tau & I_x \tau \\ 0 & Q & -\frac{I_y}{I_z}\tau & I_y \tau \\ 0 & 0 & Q - \tau & I_z \tau \\ 0 & 0 & -\tau/I_z & \tau + Q \end{bmatrix}$$

where

$$Q = \sqrt{d_x^2 + d_y^2}$$

The final simplification is possible since homology matrices are only unique up to a scale factor and we assume the eye axis center is not embedded in the projection plane.

B2 Non-Parallel Case

The following figure illustrates the distortion induced by window scaling for a head at an arbitrary position and orientation. The labeling and color scheme are the same as in Figure A-1. The user's central eye point is at **I**. **d** is the vector to the left eye. **E** is projected onto the points **H** and **G** on the projection plane. Window shifting translates points **H** and **G** by distance τ along vector **T** to points **H′** and **G′**. As discussed elsewhere [2], lines **DH′** and **AG′** do not generally intersect. Therefore, we restrict ourselves to a simple, special case where they do intersect. We restrict **A**,**D**,**d** and **E** to be contained in the XZ plane. So A_y, D_y, d_y, E_y equal zero and hence the y-coordinates of all dependent points are also zero.

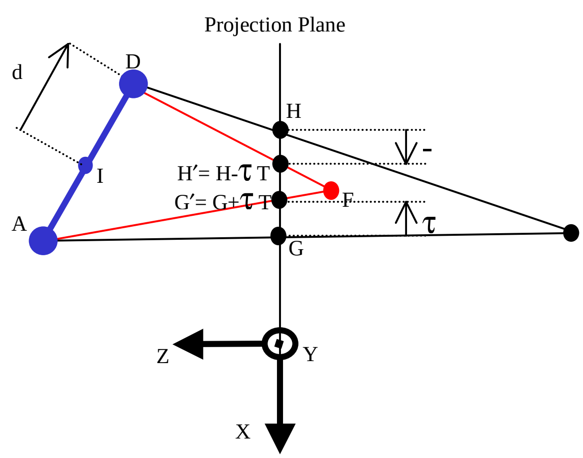

Figure B-2: This is an abstract diagram of a user viewing a stereoscopic HTD at an arbitrary orientation. See above text for details.

The derivation is parallel to section B1. When computing the expressions for *Fx* and *Fz* just remove the line 'dz=0' in the respective Mathematica files (section B1.2 and section B1.4). *Fy* is simply 0. These alterations account for the fact that we are limiting ourselves to the XZ plane and that the eye axis is no longer parallel to the screen. The resulting expression is:

$$
\begin{aligned}
\Delta_{sh}^{XZ} : F_x = \Big(\; & E_x\left(d_x d_z^3 I_x - d_x^2 d_z^2 I_z - d_x d_z I_x I_z^2 + d_x^2 I_z^3\right) + E_x^2\left(-d_x d_z^3 + d_x d_z I_z^2\right) \\
& + E_x E_z\left(d_x^2 d_z^2 - d_x^2 I_z^2 - 2|d_x| d_z^2 \tau\right) + E_z\left(2|d_x| d_z^2 I_x \tau - 2|d_x| I_x I_z^2 \tau + 2 d_x d_z I_z \tau^2\right) \\
& + E_z^2\left(d_x |d_x| d_z \tau + |d_x| I_x I_z \tau - d_x d_z \tau^2\right) \\
& + \left(d_x |d_x| d_z^3 \tau - |d_x| d_z^2 I_x I_z \tau - d_x |d_x| d_z I_z^2 \tau + |d_x| I_x I_z^3 \tau + d_x d_z^3 \tau^2 - d_x d_z I_z^2 \tau^2\right) \Big) / w
\end{aligned}
$$

$$
\begin{aligned}
F_z = -\Big(\; & E_x E_z\left(-d_x d_z\left(I_z^2 - d_z^2\right)\right) \\
& + E_z\left(I_z^2 - d_z^2\right)\left(d_x d_z I_x - d_x^2 I_z + 2|d_x| I_z \tau\right) \\
& + E_z^2\left(I_z^2 - d_z^2\right)\left(d_x^2 - |d_x| \tau\right) \\
& + \left(I_z^2 - d_z^2\right)\left(|d_x| d_z^2 \tau - |d_x| I_z^2 \tau\right) \Big) / w
\end{aligned}
\qquad (6)
$$

$$
\begin{aligned}
w = \; & E_x\left(-d_x d_z^3 + d_x d_z I_z^2\right) + E_z\left(d_x^2 d_z^2 - d_x^2 I_z^2 - 2|d_x| I_z^2 \tau\right) + E_z^2\left(|d_x| I_z \tau\right) \\
& + \left(d_x d_z^3 I_x - d_x^2 d_z^2 I_z - d_x d_z I_x I_z^2 + d_x^2 I_z^3 - |d_x| d_z^2 I_z \tau + |d_x| I_z^3 \tau\right)
\end{aligned}
$$

References

[1] Z. Wartell, L. Hodges, and W. Ribarsky. "The Analytic Distortion Induced by False-Eye Separation in Head-Tracked Stereoscopic Displays," Technical Report GIT-GVU-99-01, Graphics, Visualization and Usability Center, College of Computing, Georgia Institute of Technology, Atlanta, GA, 1999. (see **http://www.gvu.gatech.edu/gvu/reports/index.html**)

[2] Z. Wartell, L.F. Hodges, and W. Ribarsky. "A Geometric Comparison of Algorithms for Fusion Control in Stereoscopic HTDs," Technical Report GIT-GVU-01-08a, Graphics, Visualization and Usability Center, College of Computing, Georgia Institute of Technology, Atlanta, GA, 2001. (see http://www.gvu.gatech.edu/gvu/reports/index.html)

[3] Mathematica, Wolfram Research Inc., 1988-1996.