Derivation of RenderMan's Perspective Projection Matrix

Pradeep Sen

January 17, 2013

In the RenderMan specification, the perspective projection matrix takes an initial volume that is defined by the *hither* and *yon* planes, the field of view (let's call that θ_{fov}), and the frame aspect ratio (let's call that r), and transforms it into a volume that is -1/+1 in x and y and from 0 to 1 in z. In this brief writeup, we derive what the perspective transformation matrix should be to accomplish such task. To simplify things, we assume that the view volume is centered about the z axis in camera space so that $left = right$ and $top = bottom$ in Fig. 4.1 of the RenderMan spec.

First, let's examine the transformation of only the x coordinate. From a side view, this initial volume (shown in light blue) looks like this:

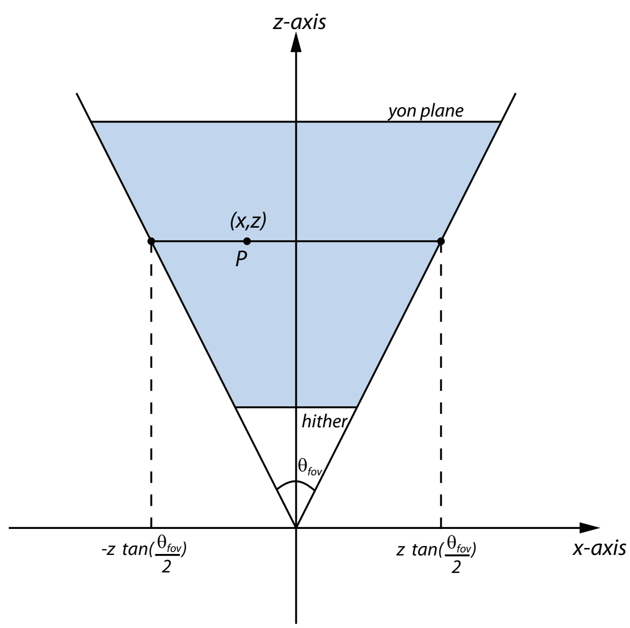

Figure 1: View volume in camera space looking down the y axis.

Our goal is to transform this trapezoidal volume into a rectangle that goes from -1 to +1 in x and from 0 to 1 in z, so that the *hither* plane gets mapped to 0 and the *yon* plane gets mapped to 1 as defined in the spec (p.25).

We can see that we can achieve this by setting our transformed coordinate x' to:

$$x' = \frac{x}{z \tan \frac{\theta_{fov}}{2}} \tag{1}$$

Likewise, we can do something similar to compute our transformed y coordinate y', the only catch here is that we have to account for the aspect ratio r. Let's call the field-of-view angle in the y direction θ_y. So we know the ratio of the two angles should be equal to the aspect ratio: $\theta_{fov}/\theta_y = r$, so $\theta_y = \theta_{fov}/r$. This makes our transformation for y:

$$y' = \frac{y}{z \tan \frac{\theta_{fov}}{2r}} \tag{2}$$

Now we want to write this as a matrix multiplication where we apply matrix M_{persp} to vector $p = [x, y, z, 1]$ in the form $p' = Mp$. However, in the equations above we have this annoying division by z which is difficult to represent in a matrix multiplication.

Fortunately for us, we are working with homogeneous coordinates, which means that we can achieve the division by z if we can set the $w' = z$. This means that our matrix-vector multiplication simply needs to compute $z \cdot x'$ and $z \cdot y$ because we will then divide by $w' = z$ to get our desired x' and y'. This means that our perspective matrix is of the form:

$$M_{persp} = \begin{bmatrix} \frac{1}{\tan(\theta_{fov}/2)} & 0 & 0 & 0 \\ 0 & \frac{1}{\tan(\theta_{fov}/2r)} & 0 & 0 \\ 0 & 0 & A & B \\ 0 & 0 & 1 & 0 \end{bmatrix} \tag{3}$$

Clearly, when we multiply this matrix by a vector $p = [x, y, z, 1]$ we will get a $w' = z$ which when homogenizing will give us our desired terms for both x' and y'. Now the only challenge is to figure out what should go into terms A and B.

We know our final z' will be of the form:

$$z' = \frac{Az + B}{z} \tag{4}$$

After multiplying our point by this matrix and homogenizing. So what should A and B be? We can figure this out by setting our boundary conditions. The spec says that when $z = hither$, $z' = 0$, and when $z = yon$ then $z' = 1$. Plugging these in we get two equations with two unknowns:

$$0 = \frac{A \cdot hither + B}{hither} \tag{5}$$
$$1 = \frac{A \cdot yon + B}{yon} \tag{6}$$

The first one gives us that $B = -A \cdot hither$, and plugging this into the second equation and solving for A gives us $A = \frac{yon}{yon - hither}$. Therefore, $B = \frac{-yon \cdot hither}{yon - hither}$. Putting all this together gives us our final perspective projection matrix:

$$M_{persp} = \begin{bmatrix} \frac{1}{\tan(\theta_{fov}/2)} & 0 & 0 & 0 \\ 0 & \frac{1}{\tan(\theta_{fov}/2r)} & 0 & 0 \\ 0 & 0 & \frac{yon}{yon - hither} & \frac{-yon \cdot hither}{yon - hither} \\ 0 & 0 & 1 & 0 \end{bmatrix} \tag{7}$$

Code this up and it should hopefully work. Let me know if there are any bugs in my derivation, I had to write this up pretty fast :)