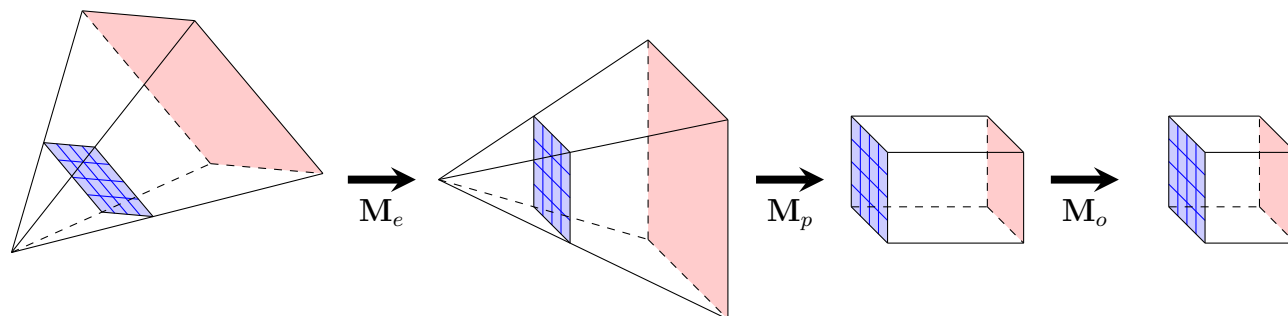


Pipeline Transformations: Projection and Camera Space

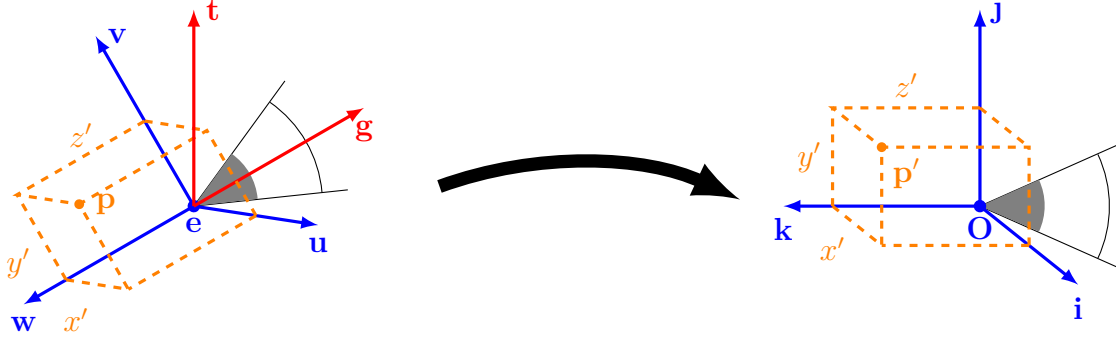
Craig Schroeder

1 Transformation stages



The goal of the transformation pipeline within a GPU is to transform primitives (points, lines, triangles) from world space into a coordinate system where they are very easy to rasterize. The viewing volume starts with a camera looking at a viewing volume, which is in the shape of a frustum. This viewing volume has a near plane (blue) and a far plane (red). The image is drawn on the near plane. The first transformation M_e translates the camera to the origin and rotates space to orient the camera along the axes. The second transformation M_p warps space to convert the frustum into a box. This decouples the components from each other. At this point, the x and y coordinates determine which pixel the point touches, and z determines the depth. The final transformation M_o scales and translates the viewing volume to the cube $-1 \leq x, y, z \leq 1$. This space is referred to as normalized device coordinates. All later stages of the rendering pipeline occur within this coordinate system.

2 Camera transformation



The input is the camera location $\mathbf{e}$, gaze direction $\mathbf{g}$, and a vector $\mathbf{t}$ establishing the orientation of the image. This vector points roughly in the upwards direction. Although $\mathbf{t}$ and $\mathbf{g}$ are directions, we do not assume that they have been normalized for us. The goal of this transformation $\mathbf{M}_e$ is to translate the camera to the origin and rotate space so the camera looks down the $-z$ axis with the image's up pointing in the y direction and the image's right pointing in the x direction.

The first step in doing this is identifying the directions that should eventually become left ($\mathbf{u}$), right ($\mathbf{v}$), and back ($\mathbf{w}$). Of these, $\mathbf{w}$ is easiest, since it points opposite $\mathbf{g}$.

$$\mathbf{w} = -\frac{\mathbf{g}}{\|\mathbf{g}\|}$$

Although we can obtain $\mathbf{v}$ by projecting $\mathbf{w}$ from $\mathbf{t}$, it is easier to obtain $\mathbf{u}$, since we know it is orthogonal to both $\mathbf{w}$ and $\mathbf{t}$.

$$\mathbf{u} = \frac{\mathbf{t} \times \mathbf{w}}{\|\mathbf{t} \times \mathbf{w}\|}$$

Finally, we can get $\mathbf{v}$ using the same approach.

$$\mathbf{v} = \mathbf{w} \times \mathbf{u}$$

Note that we must normalize $\mathbf{w}$ and $\mathbf{u}$ but not $\mathbf{v}$. This is because $\mathbf{w}$ and $\mathbf{u}$ are orthogonal and normalized, so that $\|\mathbf{v}\| = \|\mathbf{u}\|\|\mathbf{w}\|\sin\theta = 1$.

We wish to find a function describing the mapping of an arbitrary point $\mathbf{p} = (x, y, z)$ to its transformed location $\mathbf{p}' = (x', y', z')$. Note that the coordinates (x', y', z') tell us where $\mathbf{p}'$ is relative to the origin and the axes. Since we know where these axes are initially, we can use this relationship to establish where the corresponding point $\mathbf{p}$ is.

$$\begin{aligned} \mathbf{p} &= \mathbf{e} + x'\mathbf{u} + y'\mathbf{v} + z'\mathbf{w} \\ \mathbf{p} &= \underbrace{(\mathbf{u} \quad \mathbf{v} \quad \mathbf{w})}_{\mathbf{M}} \begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} + \mathbf{e} = \mathbf{M}\mathbf{p}' + \mathbf{e} \end{aligned}$$

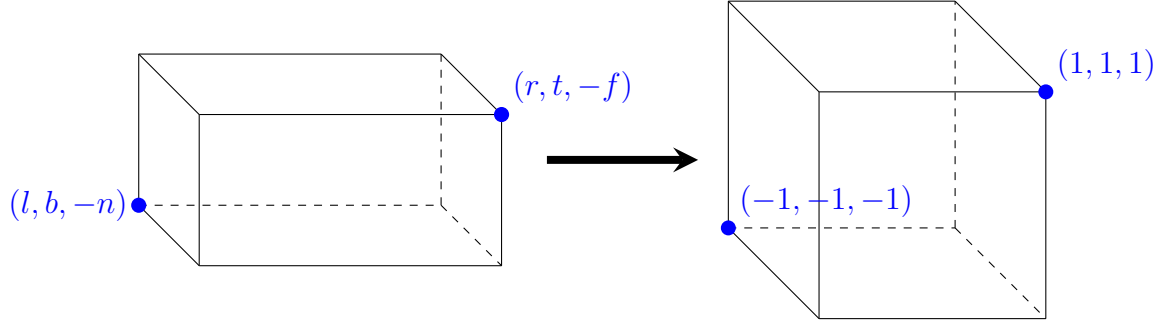
We can solve this to obtain our mapping: $\mathbf{p}' = \mathbf{M}^{-1}(\mathbf{p} - \mathbf{e})$. This calculation involves a matrix inverse, which we would like to avoid. Note that

$$\mathbf{M}^T \mathbf{M} = (\mathbf{u} \ \mathbf{v} \ \mathbf{w})^T (\mathbf{u} \ \mathbf{v} \ \mathbf{w}) = \begin{pmatrix} \mathbf{u} \cdot \mathbf{u} & \mathbf{u} \cdot \mathbf{v} & \mathbf{u} \cdot \mathbf{w} \\ \mathbf{v} \cdot \mathbf{u} & \mathbf{v} \cdot \mathbf{v} & \mathbf{v} \cdot \mathbf{w} \\ \mathbf{w} \cdot \mathbf{u} & \mathbf{w} \cdot \mathbf{v} & \mathbf{w} \cdot \mathbf{w} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} = \mathbf{I}$$

From this, Since $\mathbf{M}^T \mathbf{M} = \mathbf{I}$, $\mathbf{M}^T = \mathbf{M}^{-1}$. Thus, $\mathbf{p}' = \mathbf{M}^T(\mathbf{p} - \mathbf{e})$. This leads to the final transform matrix

$$\mathbf{M}_e = \begin{pmatrix} \mathbf{M}^T & -\mathbf{M}^T \mathbf{e} \\ \mathbf{0} & 1 \end{pmatrix}$$

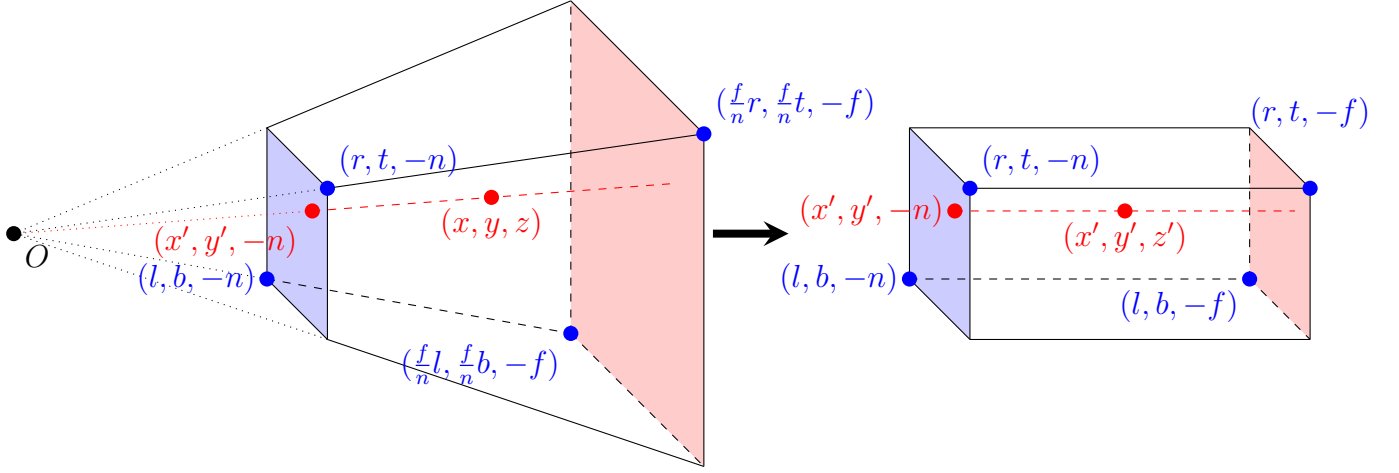
3 Orthographic transformation



This transformation maps the box with vertices $(l, b, -n)$ and $(r, t, -f)$ into normalized device coordinates, which has vertices $(-1, -1, -1)$ and $(1, 1, 1)$. Since the x, y, z directions are independent, we can consider the problem of mapping $x \in [l, r]$ to $x' \in [-1, 1]$. Let $x' = ax + b$. Then, $al + b = -1$ and $ar + b = 1$. Solving this for a, b yields $a = \frac{2}{r-l}$. and $b = -\frac{r+l}{r-l}$. The y and z directions are similar. Combining into a 3D homogeneous matrix we have

$$\mathbf{M}_o = \begin{pmatrix} \frac{2}{r-l} & 0 & 0 & -\frac{r+l}{r-l} \\ 0 & \frac{2}{t-b} & 0 & -\frac{t+b}{t-b} \\ 0 & 0 & -\frac{2}{f-n} & -\frac{f+n}{f-n} \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

4 Perspective transformation



What remains is the perspective transformation, which warps the frustum viewing volume into a box. The red line on the left maps to the red line on the right, and points on the near plane (blue) should be unchanged by the transformation. We can thus determine x' and y' by projecting the point (x, y, z) to the near plane. This yields $x' = -\frac{n}{z}x$ and $y' = -\frac{n}{z}y$.

Ideally, we would like to find a transformation such that

$$\begin{pmatrix} * & * & * & * \\ * & * & * & * \\ * & * & * & * \\ * & * & * & * \end{pmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} = \begin{pmatrix} x' \\ y' \\ z' \\ 1 \end{pmatrix} = \begin{pmatrix} -\frac{n}{z}x \\ -\frac{n}{z}y \\ * \\ 1 \end{pmatrix},$$

where $*$ represents a value that is yet to be determined. This is impossible to achieve, since x' involves division by z , which cannot happen as a result of a matrix-vector multiply. Our solution is to extend our 4D vector trick by allowing entries other than 1 as the last entry. We define

$$\begin{pmatrix} x \\ y \\ z \\ w \end{pmatrix} \equiv \begin{pmatrix} x/w \\ y/w \\ z/w \end{pmatrix}.$$

When $w = 1$, this is equivalent to what we used before. But now we have the ability to introduce a division into the final result. Note that scaling the 4D vector does not change the 3D vector that it corresponds to (as long as we scale by a positive number, but that is a detail that we can ignore for now). With this, we can scale our vector by $-\frac{z}{n}$ to clear the denominators, so that

$$\begin{pmatrix} * & * & * & * \\ * & * & * & * \\ * & * & * & * \\ * & * & * & * \end{pmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} = \begin{pmatrix} x \\ y \\ * \\ -\frac{z}{n} \end{pmatrix} \equiv \begin{pmatrix} -\frac{n}{z}x \\ -\frac{n}{z}y \\ * \\ 1 \end{pmatrix} = \begin{pmatrix} x' \\ y' \\ z' \\ 1 \end{pmatrix}.$$

Now, we have a linear problem, and many of the $*$ entries can be deduced. Although we do not yet know what the value for z' should be, we do know it should not depend on x or y . This gets us here:

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & a & b \\ 0 & 0 & -\frac{1}{n} & 0 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} = \begin{pmatrix} x \\ y \\ * \\ -\frac{z}{n} \end{pmatrix}.$$

Here, I have given names a and b to the remaining unknown entries of the matrix. From this matrix, we can actually compute z' .

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & a & b \\ 0 & 0 & -\frac{1}{n} & 0 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} = \begin{pmatrix} x \\ y \\ az + b \\ -\frac{z}{n} \end{pmatrix} \equiv \begin{pmatrix} x \\ -\frac{z}{n} \\ \frac{y}{-z/n} \\ \frac{az+b}{-z/n} \end{pmatrix} = \begin{pmatrix} x' \\ y' \\ z' \end{pmatrix}$$

Thus, $z' = \frac{az+b}{-z/n}$. This has two unknowns, so to find a and b , we must impose two constraints. We want to fix the near and far plane, so that

$$\frac{a(-n) + b}{-(-n)/n} = -n$$

$$\frac{a(-f) + b}{-(-f)/n} = -f$$

$$a(-n) + b = -n$$

$$a(-f) + b = -\frac{f^2}{n}$$

$$a(n - f) = \frac{n^2 - f^2}{n}$$

$$a = \frac{n + f}{n}$$

$$b = -n + an = f$$

$$\mathbf{M}_p = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{n+f}{n} & f \\ 0 & 0 & -\frac{1}{n} & 0 \end{pmatrix}$$