Viewing Transformations: Overview

$$
\begin{array}{ccccccc}
 & & \text{(viewport matrix)} & \text{(projection matrix)} & \text{(camera matrix)} & \text{(model matrix)} & \\
\begin{bmatrix} x_{pixel} \\ y_{pixel} \\ z_{pixel} \\ 1 \end{bmatrix} & = & M_{vp} & M_{proj} & M_{cam} & M_{model} & \begin{bmatrix} x_{model} \\ y_{model} \\ z_{model} \\ 1 \end{bmatrix} \\
\uparrow & & \uparrow & \uparrow & \uparrow & & \uparrow \\
\text{image space} & & \text{normalized device / canonical view volume} & \text{eye} & \text{world} & & \text{model coords}
\end{array}
$$

Wireframe Rendering Algorithm

Renders line segments as its primitives (not triangles) (↳ can extend to triangles later)

Input: A set of line segments $\{(a_i, b_i) : 0 \le i < n\}$

1. Form all matrices $(M_{vp}, M_{proj}, M_{cam}, M_{model})$

2. $M \leftarrow (M_{vp} \cdot M_{proj} \cdot M_{cam} \cdot M_{model})$

3. for each line segment $\vec{a}_i, \vec{b}_i$:

   $p = M\vec{a}_i$

   $q = M\vec{b}_i$

   draw_line$((x_p, y_p), (x_q, y_q))$

Viewport Matrix

- Suppose we modeled our scene in the cube $[-1,1]^3$

Input: scene in canonical view volume
normalized device coordinates

- all visible points in a cube of side length 2 centered at the origin: $(x,y,z) \in [0,1]^3$

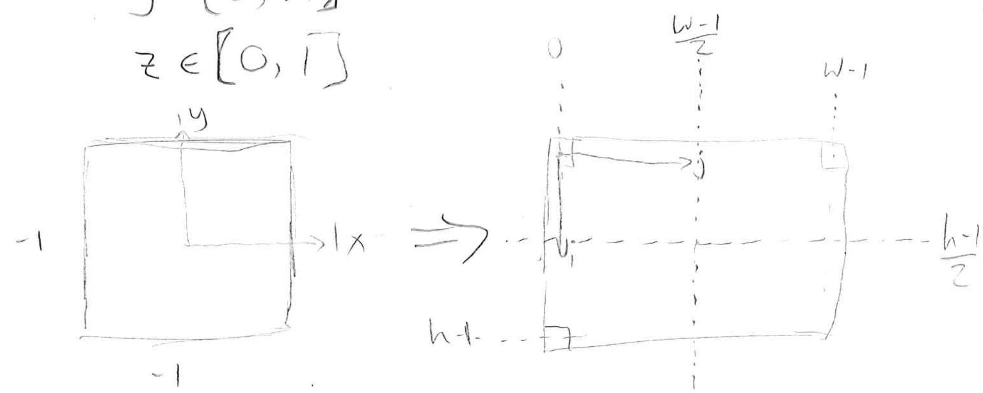
Parameters: W, H - image dimensions

Output: all visible points in pixel coordinates

$x \in [0, W-1]$

$y \in [0, H-1]$

$z \in [0, 1]$



Scale x 2 → width
y 2 → height

Translate $(0,0) \to \left(\frac{w-1}{2}, \frac{h-1}{2}\right)$

Flip y axis; leave z unchanged

Ex: write a matrix that does this

→ Why? We'll use it later to know what is in front of what.

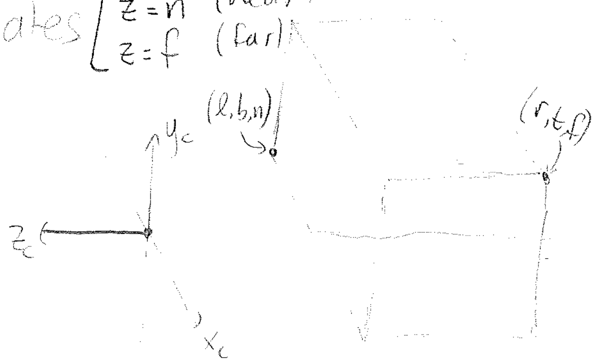
See 7.1.1 - Viewport Transformations for a solution.
Note that in the book's pixel coordinate convention, the y axis is not flipped.

Projection Matrix - Orthographic

Suppose we modeled our scene in a canonically posed orthographic camera's view area.

Input: Camera coordinates
Parameters: Orthographic viewport dimensions
Output: normalized device coordinates

$$\begin{cases} x = l & \text{(left)} \\ x = r & \text{(right)} \\ y = b & \text{(bottom)} \\ y = t & \text{(top)} \\ z = n & \text{(near)} \\ z = f & \text{(far)} \end{cases}$$



1. Translate $(l, b, n) \to (0,0,0)$

2. Scale $x: r-l \to 2$
$y: t-b \to 2$
$z: f-n \to 2$

3. Translate $(1,1,1)$ to $(0,0,0)$

$\frac{r-l}{2}$

$-l\left(\frac{2}{r-l}\right)$

$\frac{-2l}{r-l}$

$$+\left(\frac{+2l}{r-l}\right)+\frac{r-l}{r-l}=\frac{r+l}{r-l}$$

$$\begin{pmatrix} I_{3\times3} & \begin{matrix}-1\\-1\\-1\end{matrix} \\ 0 & 1 \end{pmatrix}\begin{bmatrix} \frac{2}{r-l} & 0 & 0 & 0 \\ 0 & \frac{2}{t-b} & 0 & 0 \\ 0 & 0 & \frac{2}{f-n} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}\begin{pmatrix} I_{3\times3} & \begin{matrix}-l\\-b\\-t\end{matrix} \\ 0 & 1 \end{pmatrix}$$

$$\begin{pmatrix} \frac{2}{r-l} & 0 & 0 & \frac{r+l}{r-l} \\ 0 & \frac{2}{t-b} & 0 & \frac{t+b}{t-b} \\ 0 & 0 & \frac{2}{f-n} & \frac{f+n}{f-n} \\ 0 & 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} I_{3\times3} & \begin{matrix}-1\\-1\\-1\end{matrix} \\ 0 & 1 \end{pmatrix}\begin{pmatrix} \frac{2}{r-l} & 0 & 0 & \frac{-2l}{r-l} \\ 0 & \frac{2}{t-b} & 0 & \frac{-2b}{t-b} \\ 0 & 0 & \frac{2}{f-n} & \frac{-2n}{f-n} \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Camera Matrix

Input: scene in world coordinates

Parameters: camera frame $\vec{u}, \vec{v}, \vec{w}, \vec{e}$

Output: scene in canonically-positioned camera's coords
↳ (eye at origin, looking down $-z$ axis)

$$\begin{bmatrix} \vec{u} & \vec{v} & \vec{w} & \vec{e} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$ is the frame-to-canonical matrix!

Want: canonical-to-frame, so invert it! Algebraically, or

orthonormal basis, so $Q^T = Q^{-1}$

$$\begin{bmatrix} \boxed{\vec{u} \;\; \vec{v} \;\; \vec{w}} & \vec{e} \\ 0 \;\; 0 \;\; 0 & 1 \end{bmatrix}^{-1}$$

Let $\begin{bmatrix} \vec{u} & \vec{v} & \vec{w} \end{bmatrix}_{3\times 3} = Q$

$$\downarrow$$

$$= \left(\begin{bmatrix} I_{3\times 3} & \vec{e} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} Q & 0 \\ 0 & 1 \end{bmatrix} \right)^{-1}$$

$(AB)^{-1} = B^{-1}A^{-1}$

$$= \begin{bmatrix} Q^T & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} I_{3\times 3} & -\vec{e} \\ 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} Q^T & -Q^T\vec{e} \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} \vec{u}^T & -\vec{u}^T e_x \\ \vec{v}^T & -\vec{v}^T e_y \\ \vec{w}^T & -\vec{w}^T e_z \\ 0 & 1 \end{bmatrix}$$

Model Matrix: Whatever you need to put the object where you want it!