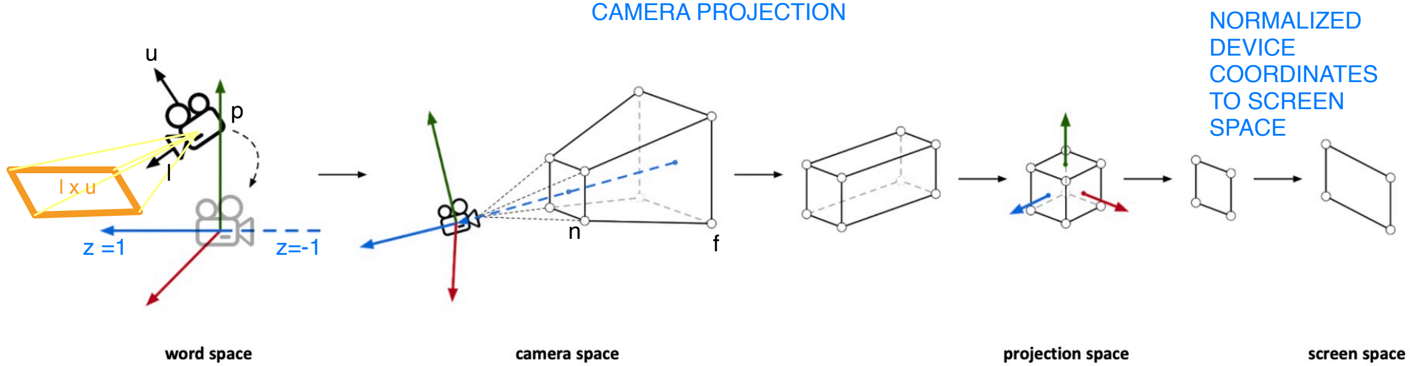
CAMERA PROJECTION



1. translate camera to origin $\begin{pmatrix} 1 & 0 & 0 & -p_1 \\ 0 & 1 & 0 & -p_2 \\ 0 & 0 & 1 & -p_3 \\ 0 & 0 & 0 & 1 \end{pmatrix}$ not a direction, just position

2. rotate $\begin{pmatrix} x_{l\times u} & x_u & x_{-l} & 0 \\ y_{l\times u} & y_u & y_{-l} & 0 \\ z_{l\times u} & z_u & z_{-l} & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$ cross product-orthogonal in the +x direction
+y is aligned with u
+z is flipped to -1

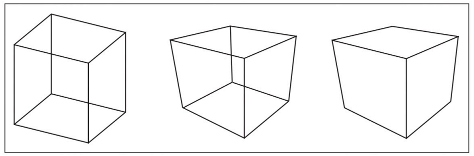


Figure 7.1. Left: wireframe cube in orthographic projection. Middle: wireframe cube in perspective projection. Right: perspective projection with hidden lines removed.

Homogenous Coordinates

[3D -> represent as 4D]

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} \rightarrow \begin{pmatrix} x \\ y \\ z \\ w \end{pmatrix} \Leftrightarrow \begin{pmatrix} x/w \\ y/w \\ z/w \end{pmatrix}$$

Homogenous Form of translation

$$= \begin{pmatrix} 1 & 0 & 0 & tx \\ 0 & 1 & 0 & ty \\ 0 & 0 & 1 & tz \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}$$

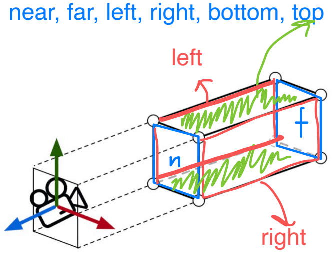
Homogenous Form of scaling

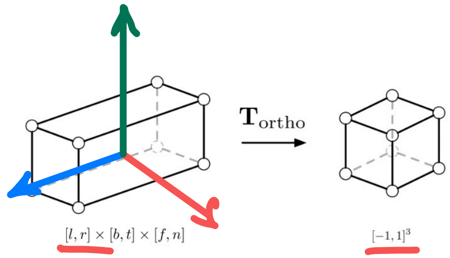
$$= \begin{pmatrix} sx & 0 & 0 & 0 \\ 0 & sy & 0 & 0 \\ 0 & 0 & sz & 0 \\ 0 & 0 & 0 & sw \end{pmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}$$

ORTHOGRAPHIC PROJECTION

near, far, left, right, bottom, top

first we translate the center of the cube to the origin
-and then scale it's length, height, width to 2 to get the unit circle





the parallelogram is along the -z axis (we assume the viewer is looking along the -z axis with his head pointing in the y direction => implies: n>f (because n is closer to the viewer)

because we're looking to center it/middle e.g. (4+3)/2 =3.5

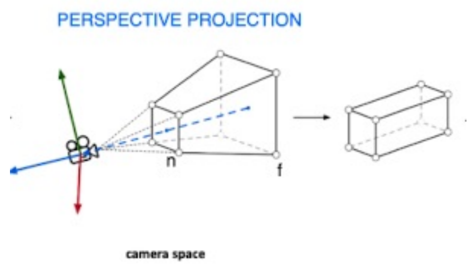
translation vector

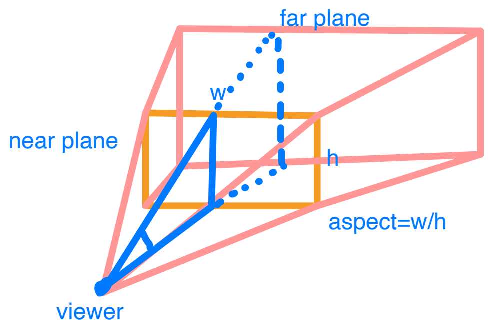
our interval for the unit cube [-1,1] defined for [l,r]
=:> l = -1, r=1
r-l = 1-(-1)=2
2:2=1 (unit)

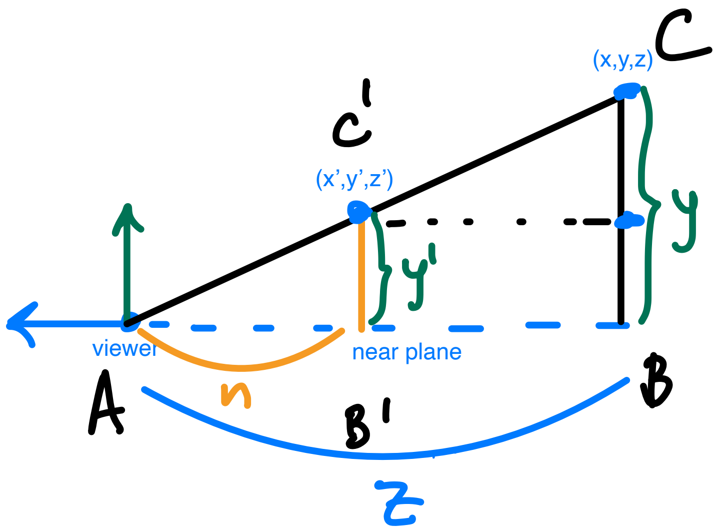
left and right are along the x axis
top and bottom are along the y axis
near far are along the z axis

$$\mathbf{T}_{\text{ortho}} = \underbrace{\begin{pmatrix} 2/(r-l) & 0 & 0 & 0 \\ 0 & 2/(t-b) & 0 & 0 \\ 0 & 0 & 2/(n-f) & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}}_{\text{scale}} \underbrace{\begin{pmatrix} 1 & 0 & 0 & -(r+l)/2 \\ 0 & 1 & 0 & -(t+b)/2 \\ 0 & 0 & 1 & -(n+f)/2 \\ 0 & 0 & 0 & 1 \end{pmatrix}}_{\text{translation}}$$

$$= \begin{pmatrix} 2/(r-l) & 0 & 0 & (l+r)/(l-r) \\ 0 & 2/(t-b) & 0 & (b+t)/(b-t) \\ 0 & 0 & 2/(n-f) & (f+n)/(f-n) \\ 0 & 0 & 0 & 1 \end{pmatrix}$$







$$\frac{BC}{B'C'} = \frac{AB}{AB'} \Rightarrow \frac{y'}{y} = \frac{n}{z}$$

$$\Rightarrow y' = \frac{ny}{z} \qquad x' = \frac{nx}{z}$$

homogenous coordinates:

but: this division by z in the x and y it's imposible to achieve just by multiplication

We know:
=> homo coords $\begin{pmatrix} x \\ y \\ z \\ w \end{pmatrix} \Leftrightarrow \begin{pmatrix} x/w \\ y/w \\ z/w \end{pmatrix}$

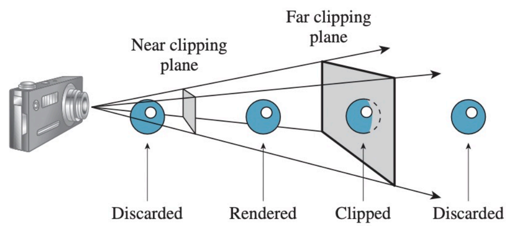


Figure 13.5: Objects outside the view frustum will not be rendered.

We want to transform the view frustum with a perspective matrix into the orthographic view volume

projected coords

$$\begin{pmatrix} & ? & \end{pmatrix} \cdot \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} = \begin{pmatrix} nx/z \\ ny/z \\ z \\ 1 \end{pmatrix} = \begin{pmatrix} nx \\ ny \\ z^2 \\ z \end{pmatrix}$$

position vector

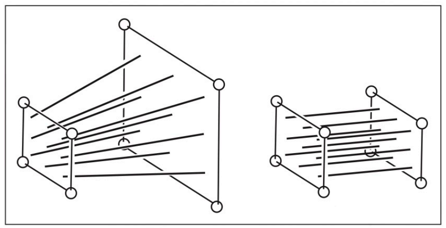


Figure 7.13. The perspective projection maps any line through the origin/eye to a line parallel to the z-axis and without moving the point on the line at $z = n$.

• near plane : $z = n \Rightarrow$ map lines through the origin/eye to a line parallel to the z-axis, but keep the point at line z=n unchanged (center/ blue dot is fixed)

$$\begin{pmatrix} n & 0 & 0 & 0 \\ 0 & n & 0 & 0 \\ 0 & 0 & w_1 & w_2 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} x \\ y \\ n \\ 1 \end{pmatrix} = \begin{pmatrix} nx \\ ny \\ n^2 \\ n \end{pmatrix} \Rightarrow n.w_1 + w_2 = n^2 \; (1)$$

$$\_.x + \_.y + \_.n + \_.1 = n^2$$

$$\_.x + \_.y + \_.n + \_.1 = n$$

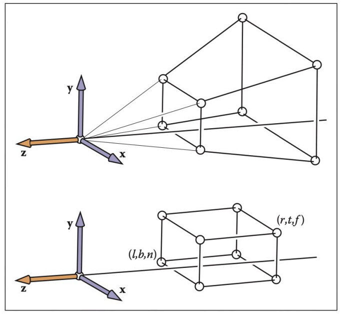


Figure 7.12. The perspective projection leaves points on the $z = n$ plane unchanged and maps the large $z = f$ rectangle at the back of the perspective volume to the small $z = f$ rectangle at the back of the orthographic volume.

- far plane. $z=f$

leave points on the near plane unchanged (zeros) and map the far plane (z=f) at the back of the orthographic volume

$$\begin{pmatrix} n & 0 & 0 & 0 \\ 0 & n & 0 & 0 \\ 0 & 0 & w_1 & w_2 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} 0 \\ 0 \\ f \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ f^2 \\ f \end{pmatrix} \Rightarrow f \cdot w_1 + w_2 = f^2 \quad (2)$$

from (1) and (2):

$$\left| \begin{array}{l} n w_1 + w_2 = n^2 \\ f w_1 + w_2 = f^2 \end{array} \right. \qquad \left| \begin{array}{l} w_2 = n^2 - n w_1 \\ w_2 = f^2 - f w_1 \end{array} \right.$$

$$\rightarrow n^2 - n w_1 = f^2 - f w_1$$

$$f w_1 - n w_1 = f^2 - n^2$$

$$w_1 \cancel{(f-n)} = \cancel{(f-n)}(f+n)$$

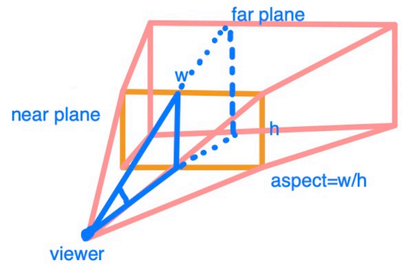
$$w_1 = f+n$$

$$\left| \begin{array}{l} w_1 = f+n \\ w_2 = n^2 - n(f+n) \end{array} \right. \rightarrow w_2 = \cancel{n^2} - nf - \cancel{n^2}$$

$$w_2 = -nf$$

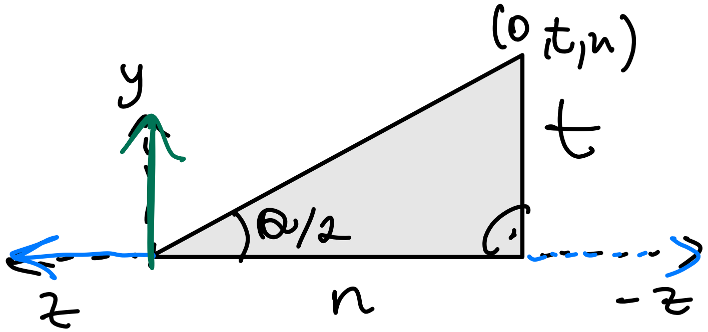
$$T_{persp \to ort} \Rightarrow \begin{pmatrix} n & 0 & 0 & 0 \\ 0 & n & 0 & 0 \\ 0 & 0 & n+f & -nf \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

Perspective Matrix = $T_{ort} * T_{persp \to ort}$



$$\Rightarrow \quad \mathbf{T}_{\text{ortho}}\mathbf{T}_{\text{persp}\to\text{ortho}} = \begin{pmatrix} 2n/(r-l) & 0 & (l+r)/(l-r) & 0 \\ 0 & 2n/(t-b) & (b+t)/(b-t) & 0 \\ 0 & 0 & (n+f)/(n-f) & 2nf/(f-n) \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

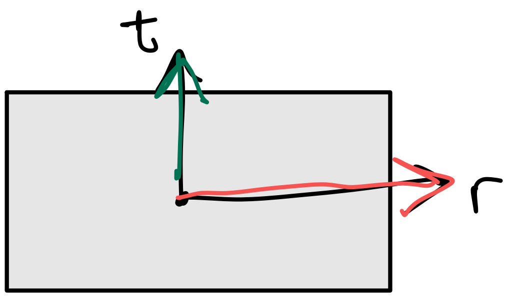
- now : what are l, r, b, t ?



symmetric projections both vertically & horizontally

$\Rightarrow \underline{t} = \underline{-n \tan \theta/2}$ (we're looking along the -z axis)

$\underline{b = -t} = -(-n)\tan\theta/2 = \underline{n \tan \theta/2}$



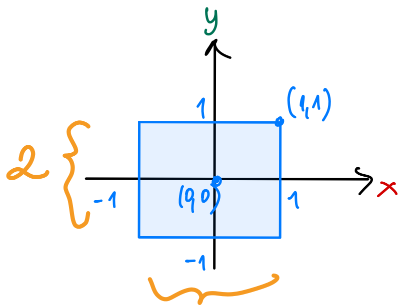
$\Rightarrow \underset{\text{aspect}}{\lambda} = \dfrac{r}{t} \Rightarrow r = \lambda t$

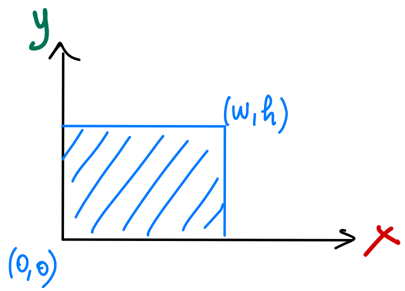
$\underline{l = -r} = -\lambda t = -\lambda(-n)\tan\theta/2 = \lambda n \tan\theta/2$

Last Step: Screen Transformation $[-1,1]^2 \to [0,w] \times [0,h]$

└ from 2D viewing plane to pixel coordinates

└ take points & transform into a $w \times h$ pixel image





$$T_{viewport} = \underbrace{\begin{pmatrix} s_x & 0 & 0 & 0 \\ 0 & s_y & 0 & 0 \\ 0 & 0 & s_z & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}}_{\text{scale}} \underbrace{\begin{pmatrix} 1 & 0 & 0 & t_x \\ 0 & 1 & 0 & t_y \\ 0 & 0 & 1 & t_z \\ 0 & 0 & 0 & 1 \end{pmatrix}}_{\text{translate}} = \begin{pmatrix} s_x & 0 & 0 & s_x t_x \\ 0 & s_y & 0 & s_y t_y \\ 0 & 0 & s_z & s_z t_z \\ 0 & 0 & 0 & 1 \end{pmatrix} \; t = \begin{pmatrix} w/2 & 0 & 0 & w/2 \\ 0 & h/2 & 0 & h/2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

1) translate the 2x2 square $\Rightarrow t_x = t_y = 1$ / z is irrelevant $\Rightarrow t_z = 0$ /

2) scale the width & height $\Rightarrow s_x = w/2$ / z is irrelevant $\Rightarrow s_z = 1$ /

$s_y = h/2$