CPSC 314
Computer Graphics

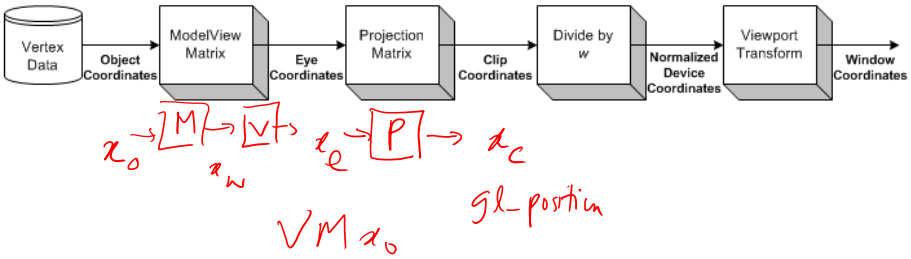
Dinesh K. Pai

L14
Projection and Depth

Today

- Announcements
  - My office hour is each W 4-5pm in the lab (005)
  - Signup for the next office hour by W 12:00
    https://tinyurl.com/CS314-OfficeHour
  - Reminder: Lecture notes are available on the course web page http://sensorimotor.cs.ubc.ca/cpsc-314/
    (under "Lectures" tab)
- Lecture
  - Projection and Depth

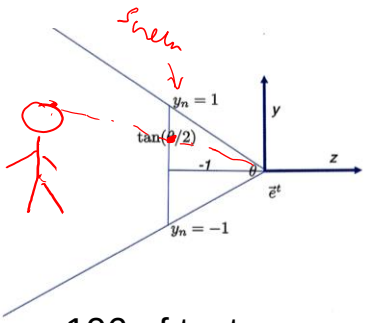
Review: OpenGL pipeline



http://www.songho.ca/opengl/gl_transform.html

3

PerspectiveCamera
Eye coords → Clip coords



$$\begin{bmatrix} \frac{1}{a\tan\left(\frac{q}{2}\right)} & 0 & 0 & 0 \\ 0 & \frac{1}{\tan\left(\frac{q}{2}\right)} & 0 & 0 \\ 0 & 0 & \frac{f+n}{f-n} & -\frac{2fn}{f-n} \\ 0 & 0 & -1 & 0 \end{bmatrix}$$

See p. 106 of text

gluPerspective (which uses glFrustum) in OpenGL

also

https://threejs.org/docs/#api/cameras/PerspectiveCamera

Today: where does this come from?

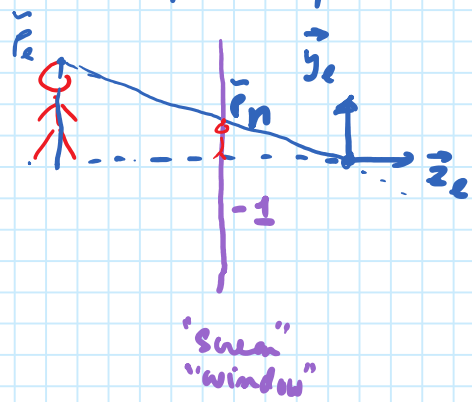
Caveat: In Book near and far are negative. In API's near & far are positive

4

October 11, 2017 10:16 AM

Basic Geometry of Perspective (L13)

$\bar{p}_e = \begin{pmatrix} y \\ z \\ 1 \end{pmatrix}$



x will behave similar to y

$\bar{p}_n = \begin{pmatrix} y_n \\ z_n \\ 1 \end{pmatrix} = ?$

$z_n = -1$ by design

$y_n = -\dfrac{y_n}{z_n}$ because $\dfrac{y_n}{y_e} = \dfrac{-1}{z_e}$

In Vector form (writing $\bar{p}_e$ as $\vec{p}$)

$$\begin{pmatrix} y \\ z \\ 1 \end{pmatrix} \longrightarrow \begin{pmatrix} -y/z \\ -1 \\ 1 \end{pmatrix} = -\frac{1}{z}\begin{pmatrix} y \\ z \\ -z \end{pmatrix} \equiv \begin{pmatrix} y \\ z \\ -z \end{pmatrix}$$

So projection achieved by just changing "w"!! in homog. coords

☆ General Principle:

Identify all non-zero multiples of homogeneous coordinates of a point with itself

eg. $\begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} \equiv \begin{pmatrix} 25x \\ 25y \\ 25z \\ 25 \end{pmatrix} \xrightarrow[\text{also called Homogenize or "perspective divide"}]{\text{Divide by the "w"}} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}$

multiplier is easy to see

So we can delay the perspective division (expensive) until geometry has been clipped to the View Frustum

Using this new insight into homogeneous coordinates we can write the non-linear perspective transform as a matrix

Attempt 1:

$$P_1 = \left[\begin{array}{cc|c} 1 & \cdot & \cdot \\ \cdot & 1 & \cdot \\ \hline \cdot & -1 & \cdot \end{array}\right]$$

✓ Doesn't even require multiplication, just swizzling

want

$$\begin{pmatrix} y \\ z \\ 1 \end{pmatrix} \rightarrow \begin{pmatrix} y \\ z \\ -z \end{pmatrix}$$

Even though this achieves projection, it loses all depth information 3D → 2D

Attempt 2: Projective Transformation 3D → 3D

$$P_u = \left[\begin{array}{cc|c} 1 & \cdot & \cdot \\ \cdot & \cdot & 1 \\ \hline & -1 & \cdot \end{array}\right]$$

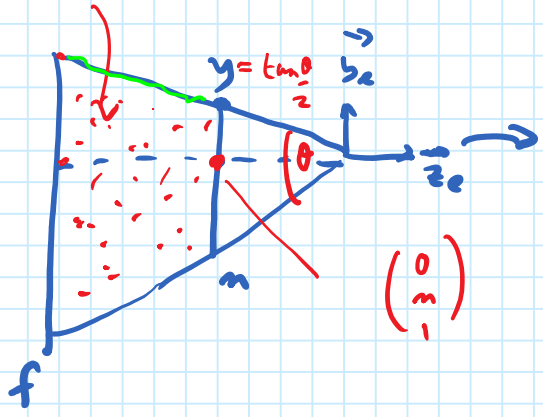
Unhinging Transform

$$\begin{pmatrix} y \\ z \\ 1 \end{pmatrix} \rightarrow \begin{pmatrix} y \\ 1 \\ -z \end{pmatrix} \xrightarrow{\text{Perspective divide}} \begin{pmatrix} -y/z \\ -1/z \\ 1 \end{pmatrix}$$

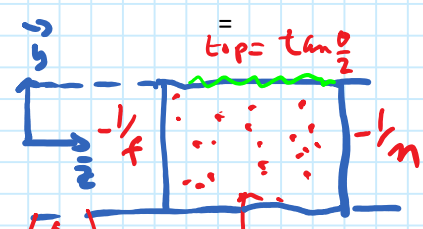
depth like

Can be used for hidden surface removal (Ch. 11)

View frustum



$$\begin{pmatrix} 0 \\ n \\ 1 \end{pmatrix} \rightarrow \begin{pmatrix} 0 \\ 1 \\ -n \end{pmatrix} = \begin{pmatrix} 0 \\ -1/n \\ 1 \end{pmatrix}$$



transformed view frustum