CSE 167: Introduction to Computer Graphics Lecture #3: Projection

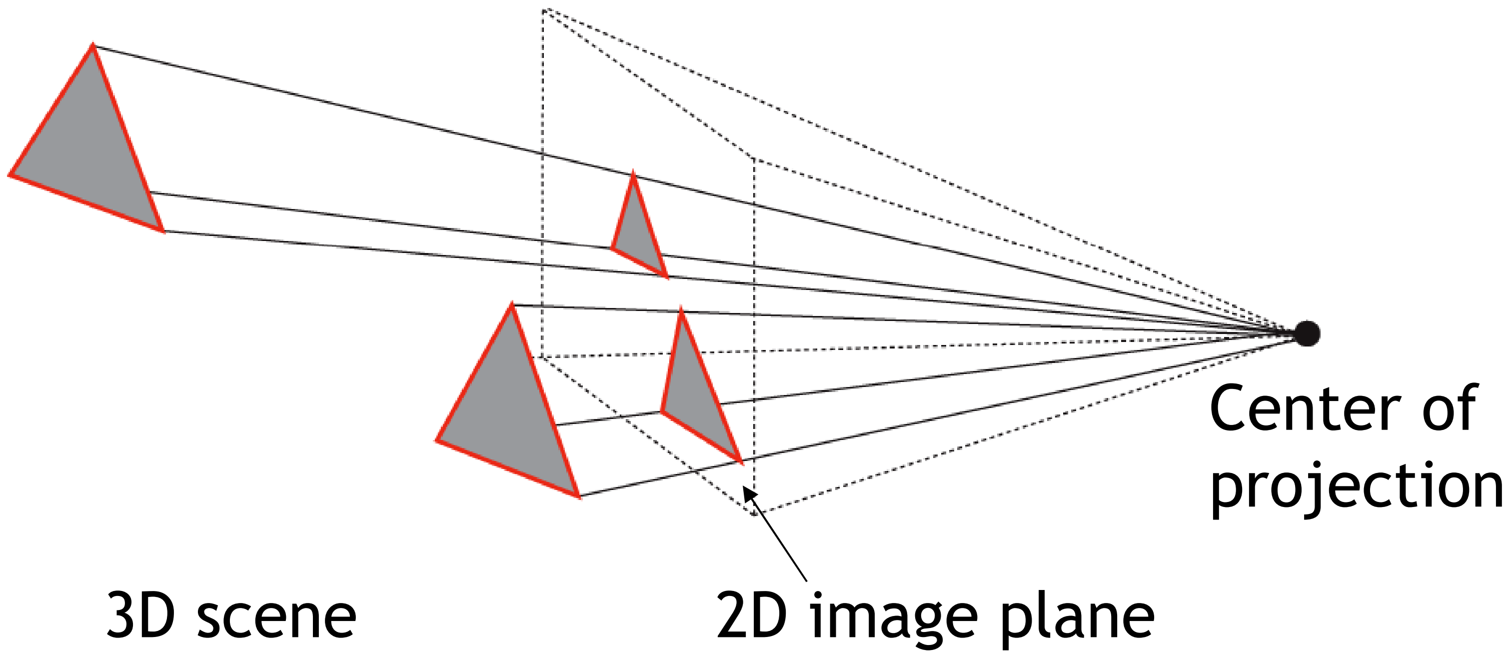
Jürgen P. Schulze, Ph.D.
University of California, San Diego
Fall Quarter 2011

Perspective Projection

- Most common for computer graphics
- Simplified model of human eye, or camera lens (*pinhole camera*)
- Things farther away appear to be smaller
- Discovery attributed to Filippo Brunelleschi (Italian architect) in the early 1400's

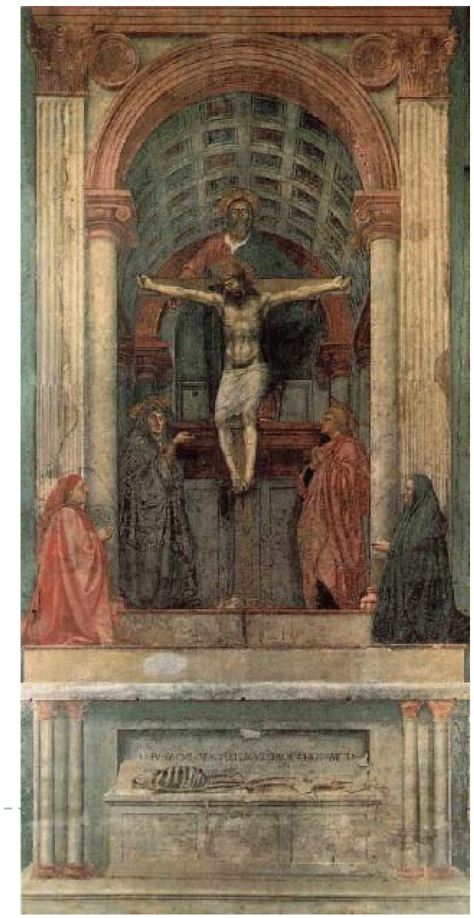
Perspective Projection

- Project along rays that converge in center of projection



Perspective Projection

Parallel lines are
no longer parallel,
converge in one point



Earliest example:
La Trinitá (1427) by Masaccio

Perspective Projection

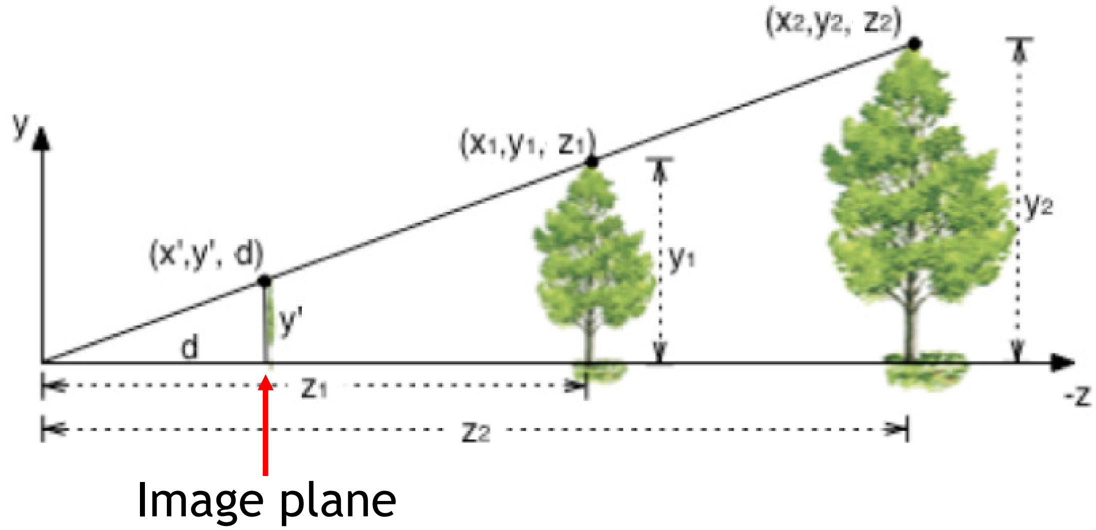
The math: simplified case

$$\frac{y'}{d} = \frac{y_1}{z_1}$$

$$y' = \frac{y_1 d}{z_1}$$

$$x' = \frac{x_1 d}{z_1}$$

$$z' = d$$



Perspective Projection

The math: simplified case

$$x' = \frac{x_1 d}{z_1}$$

$$y' = \frac{y_1 d}{z_1}$$

$$z' = d$$

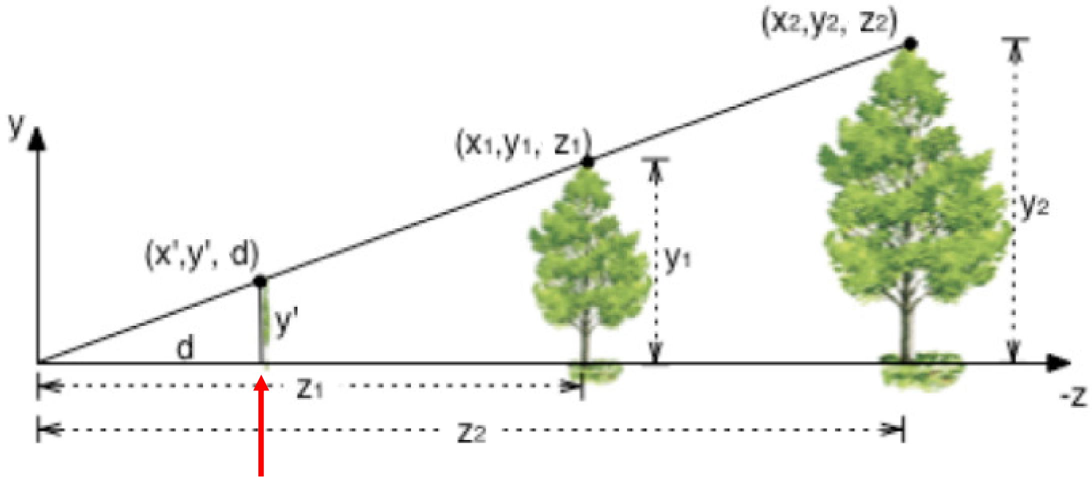


Image plane

- We can express this using homogeneous coordinates and 4x4 matrices

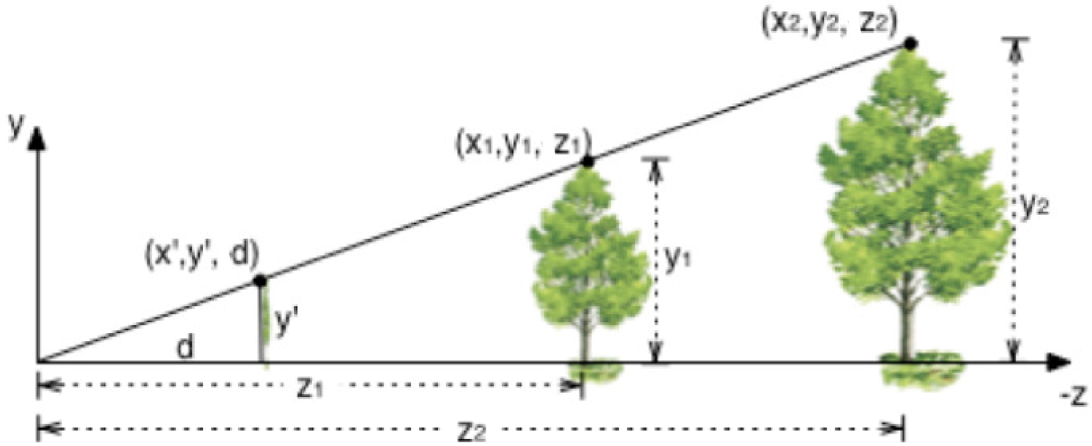
Perspective Projection

The math: simplified case

$$x' = \frac{x_1 d}{z_1}$$

$$y' = \frac{y_1 d}{z_1}$$

$$z' = d$$



$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1/d & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} x \\ y \\ z \\ z/d \end{bmatrix} \Rightarrow \begin{bmatrix} xd/z \\ yd/z \\ d \\ 1 \end{bmatrix}$$

Projection matrix Homogeneous division

Perspective Projection

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1/d & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} x \\ y \\ z \\ z/d \end{bmatrix} = \begin{bmatrix} xd/z \\ yd/z \\ d \\ 1 \end{bmatrix}$$

Projection matrix Homogeneous division

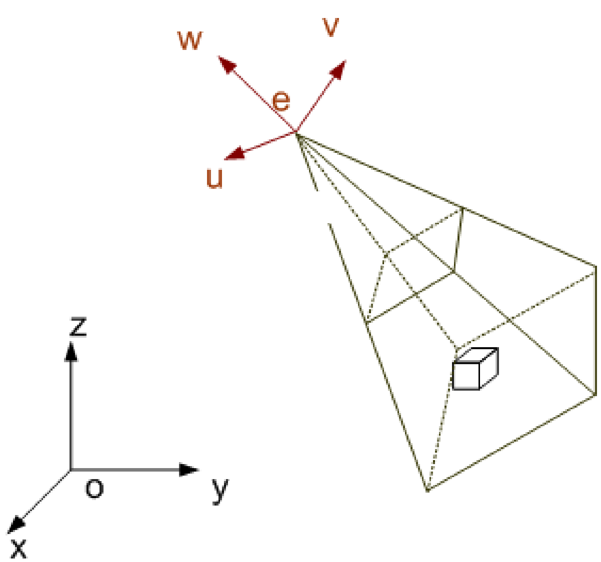
- Using projection matrix, homogeneous division seems more complicated than just multiplying all coordinates by d/z, so why do it?
- It will allow us to:
  - handle different types of projections in a unified way
  - define arbitrary view volumes
- Divide by *w* (perspective division, homogeneous division) after performing projection transform
  - Graphics hardware does this automatically

View Volumes

- Define 3D volume seen by camera

Perspective view volume

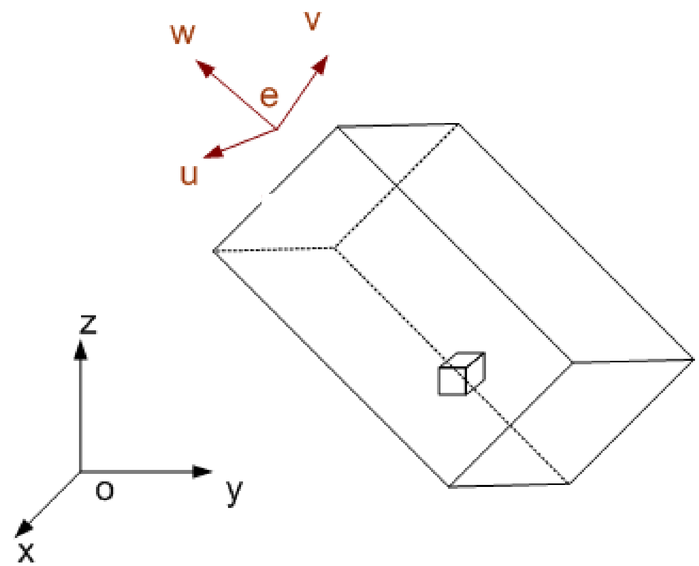
Camera coordinates



World coordinates

Orthographic view volume

Camera coordinates

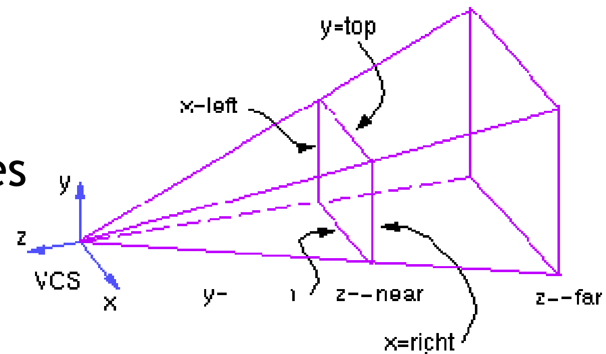


World coordinates

Perspective View Volume

General view volume

Camera
coordinates



- Defined by 6 parameters, in camera coordinates
  - Left, right, top, bottom boundaries
  - Near, far clipping planes
- Clipping planes to avoid numerical problems
  - Divide by zero
  - Low precision for distant objects
- Usually symmetric, i.e., left=-right, top=-bottom