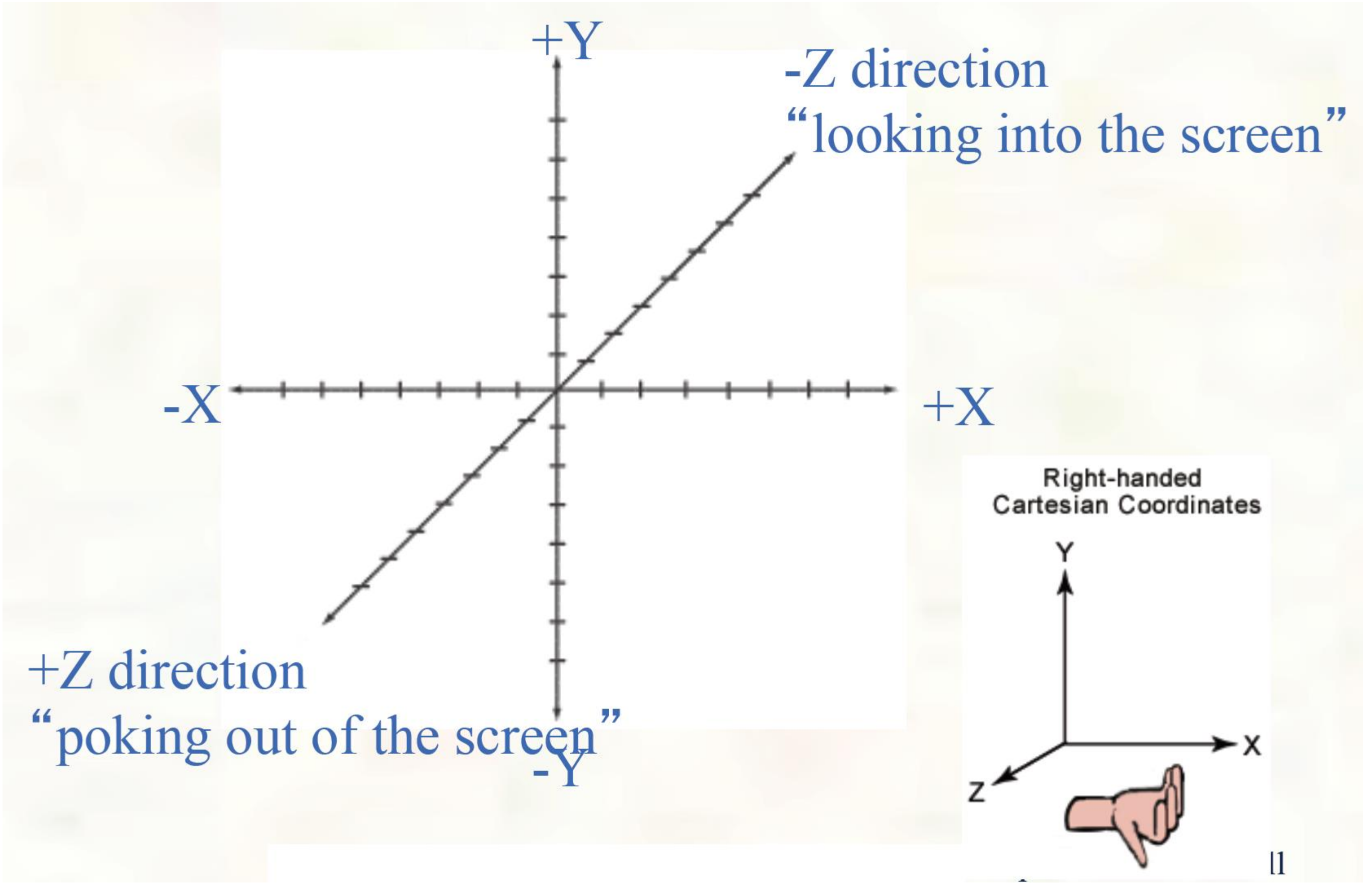
Slide Credit: Donald S. Fussell

CS354 Computer Graphics
Viewing and Projections

Qixing Huang

February 19th 2018

Eye Coordinates (not NDC)

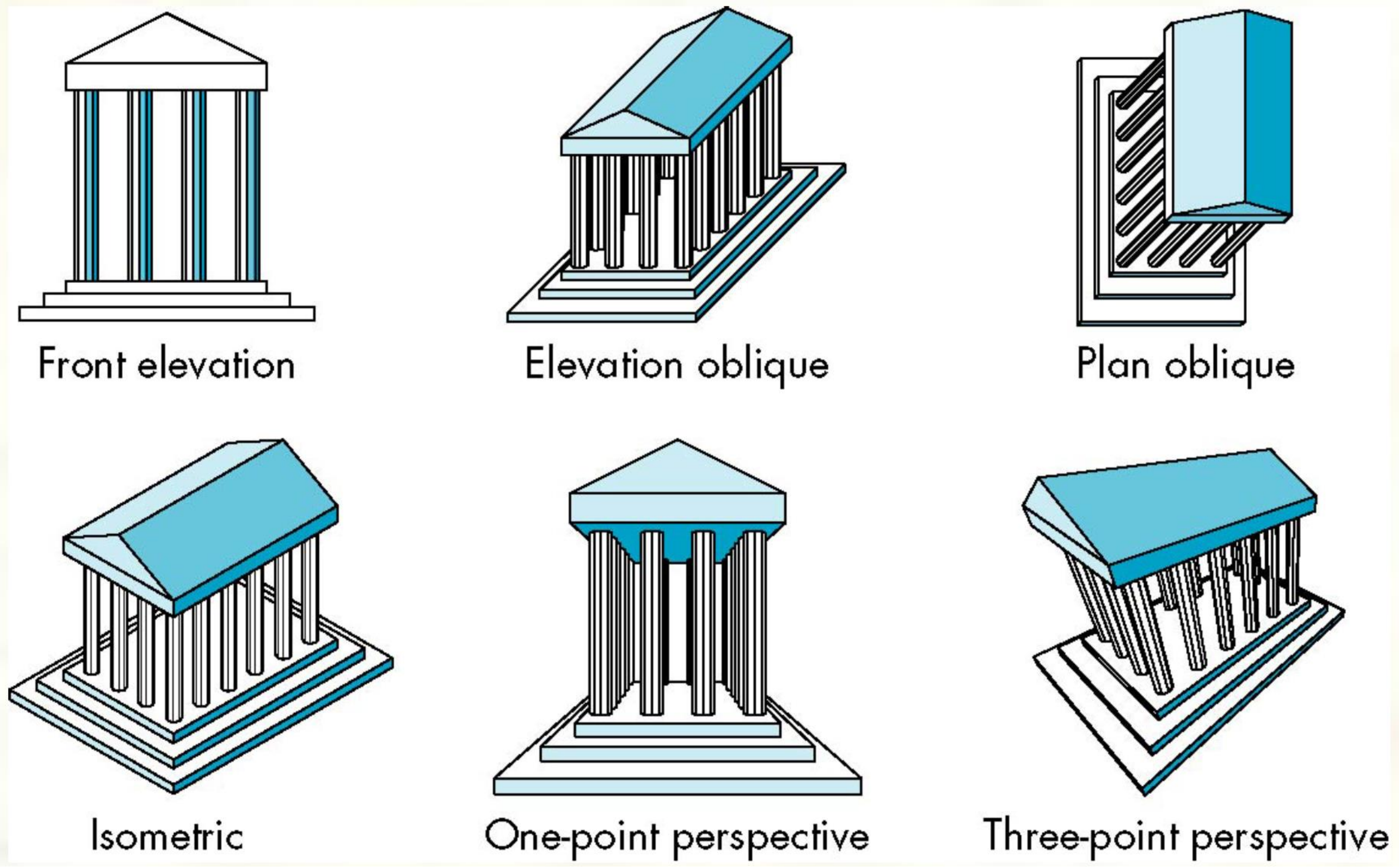


Planar Geometric Projections

- Standard projections project onto a plane
- Projectors are lines that either
  - converge at a center of projection
  - are parallel

- Such projections preserve lines
  - but not necessarily angles

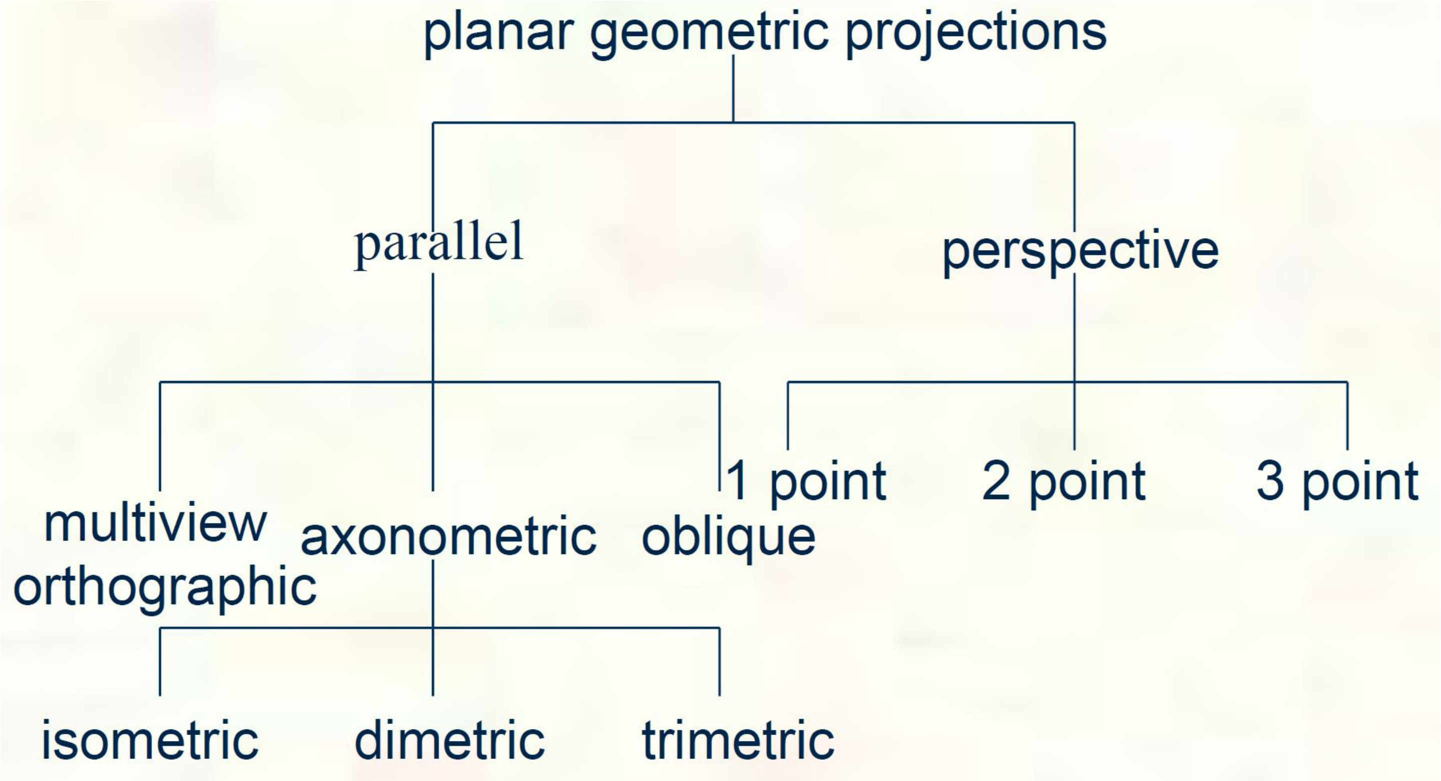
Classical Projections



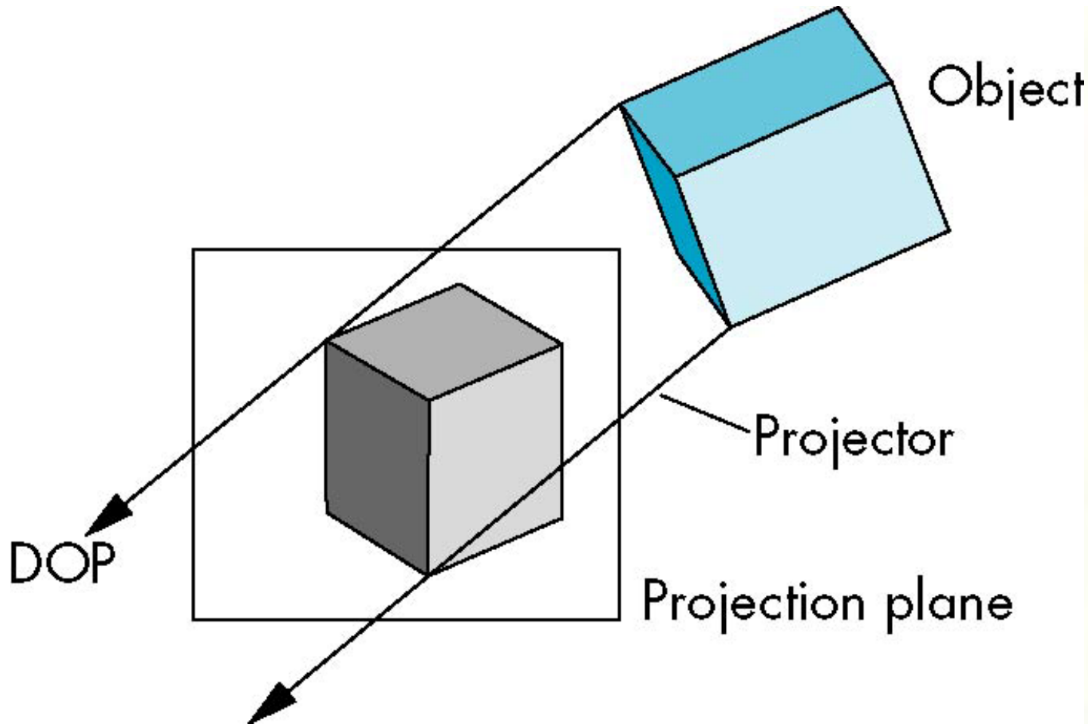
Perspective vs Parallel

- Computer graphics treats all projections the same and implements them with a single pipeline
- Classical viewing developed different techniques for drawing each type of projection
- Fundamental distinction is between parallel and perspective viewing even though mathematically parallel viewing is the limit of perspective viewing

Taxonomy of Projections

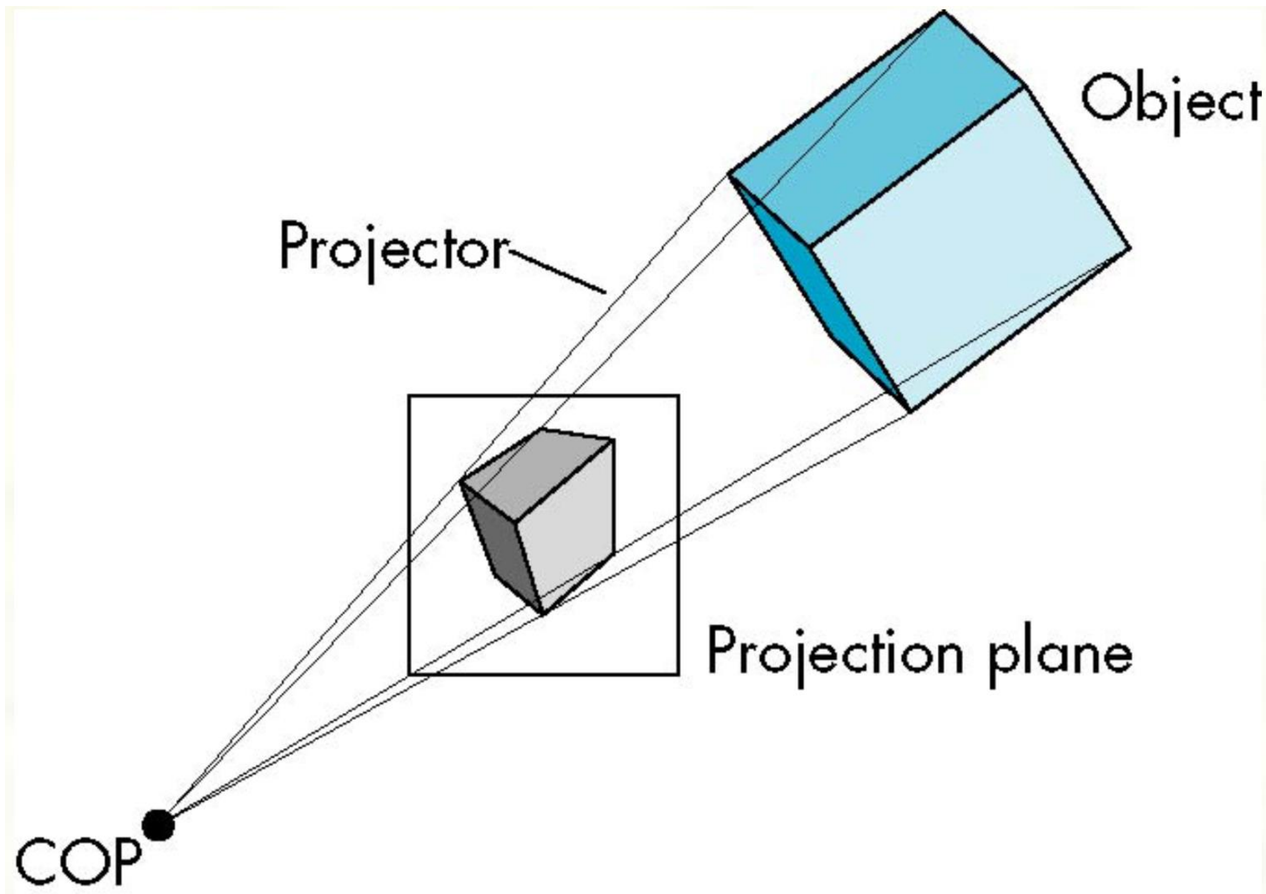


Parallel Projection



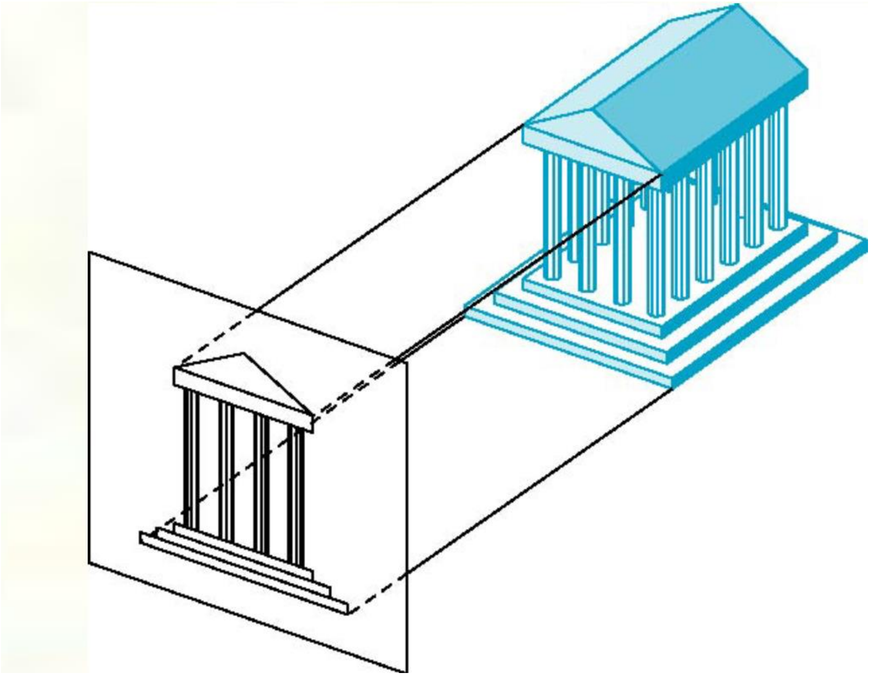
A parallel projection is a projection of an object in three-dimensional space onto a fixed plane, known as the projection plane or image plane, where the rays, known as lines of sight or projection lines, are parallel to each other.

Perspective Projection



Orthographic Projection

Projectors are orthogonal to projection surface



Orthographic projection is a means of representing three-dimensional objects in two dimensions. It is a form of parallel projection, in which all the projection lines are orthogonal to the projection plane.

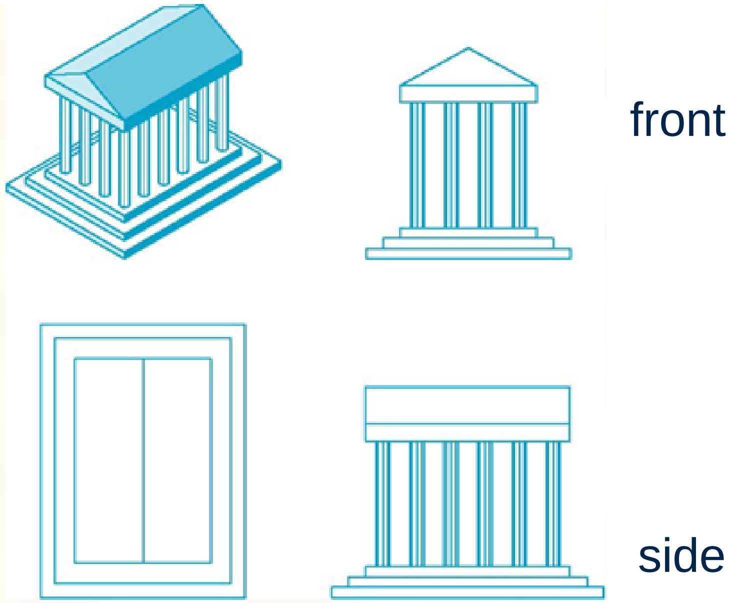
Multiview Orthographic Projection

Projection plane parallel to principal face

Usually form front, top, side views

isometric (not multiview orthographic view)

in CAD and architecture, we often display three multiviews plus isometric



Isometric projection is a method for visually representing three-dimensional objects in two dimensions. It is an axonometric projection in which the three coordinate axes appear equally foreshortened and the angle between any two of them is 120 degrees.

Multiview Orthographic Projection

- Preserves both distances and angles
  - Shapes preserved
  - Can be used for measurements
    - Building plans
    - Manuals

- Cannot see what object really looks like because many surfaces hidden from view
  - Often we add the isometric

Projections and Normalization

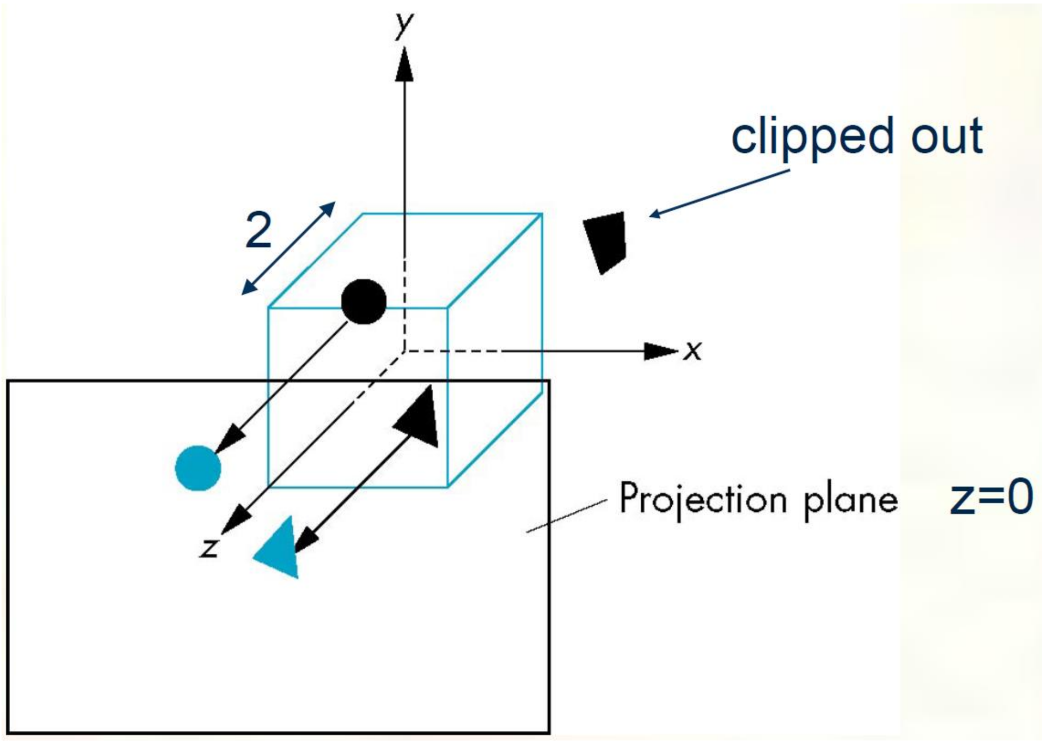
- The default projection in the eye (camera) frame is orthogonal
- For points within the default view volume

$$x_p = x$$
$$y_p = y$$
$$z_p = 0$$

- Most graphics systems use *view normalization*
  – All other views are converted to the default view by transformations that determine the projection matrix
  – Allows use of the same pipeline for all views

Default Projection

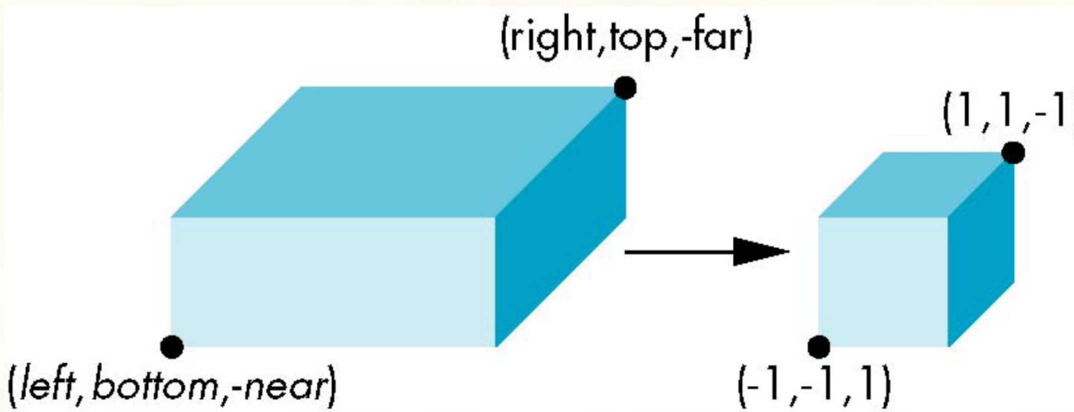
Default projection is orthographic



Orthogonal Normalization

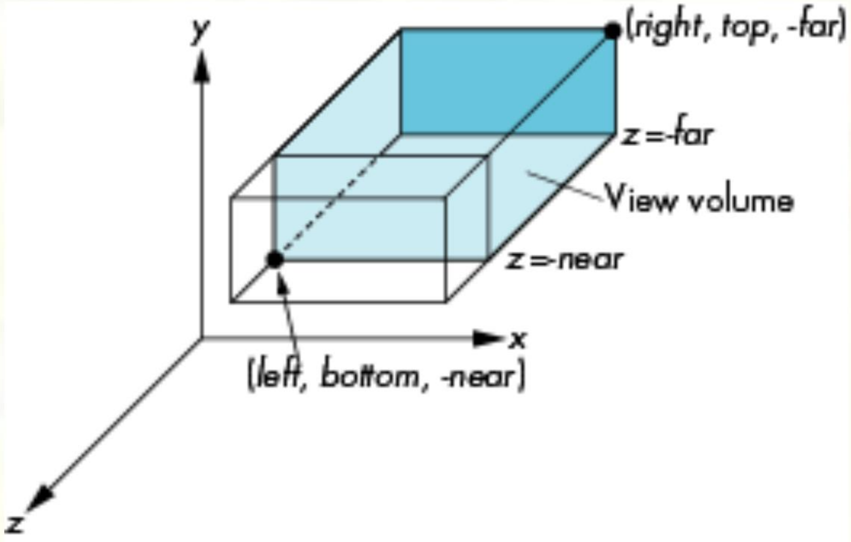
```
glOrtho(left,right,bottom,top,near,far)
```

normalization ⇒ find transformation to convert specified clipping volume to default



OpenGL Orthogonal Viewing

glOrtho(left,right,bottom,top,near,far)



near and **far** measured from camera

Homogeneous Representation

default orthographic projection

$$x_p = x$$
$$y_p = y$$
$$z_p = 0$$
$$w_p = 1$$

$$\mathbf{p}_p = \mathbf{M}\mathbf{p}$$

$$\mathbf{M} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

In practice, we can let $\mathbf{M} = \mathbf{I}$ and set the z term to zero later

https://en.wikipedia.org/wiki/Homogeneous_coordinates

Orthographic Eye to NDC

- Two steps
  - Move center to origin

    T(-(left+right)/2, -(bottom+top)/2,-(near+far)/2))
  - Scale to have sides of length 2

    S(2/(left-right),2/(top-bottom),2/(near-far))

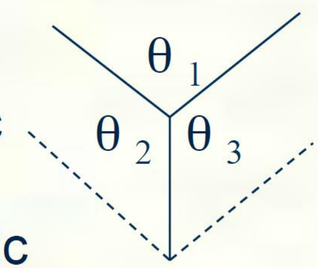
$$\mathbf{P} = \mathbf{ST} = \begin{bmatrix} \frac{2}{right-left} & 0 & 0 & -\frac{right+left}{right-left} \\ 0 & \frac{2}{top-bottom} & 0 & -\frac{top+bottom}{top-bottom} \\ 0 & 0 & \frac{2}{near-far} & -\frac{far+near}{far-near} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

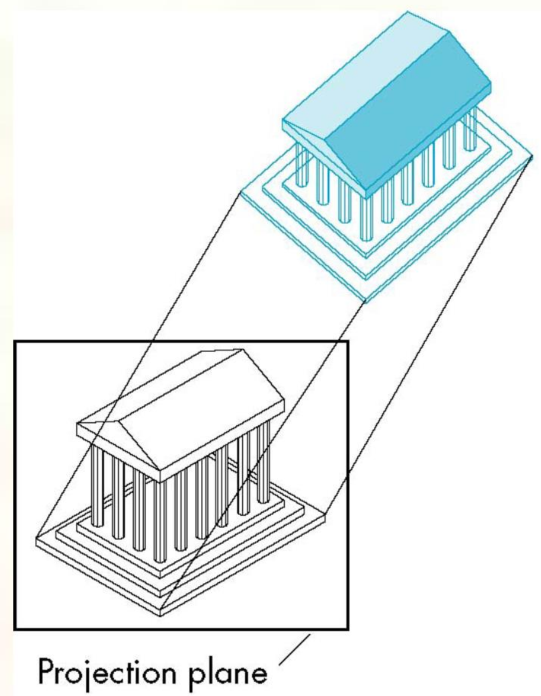
Axonometric Projections

Allow projection plane to move relative to object

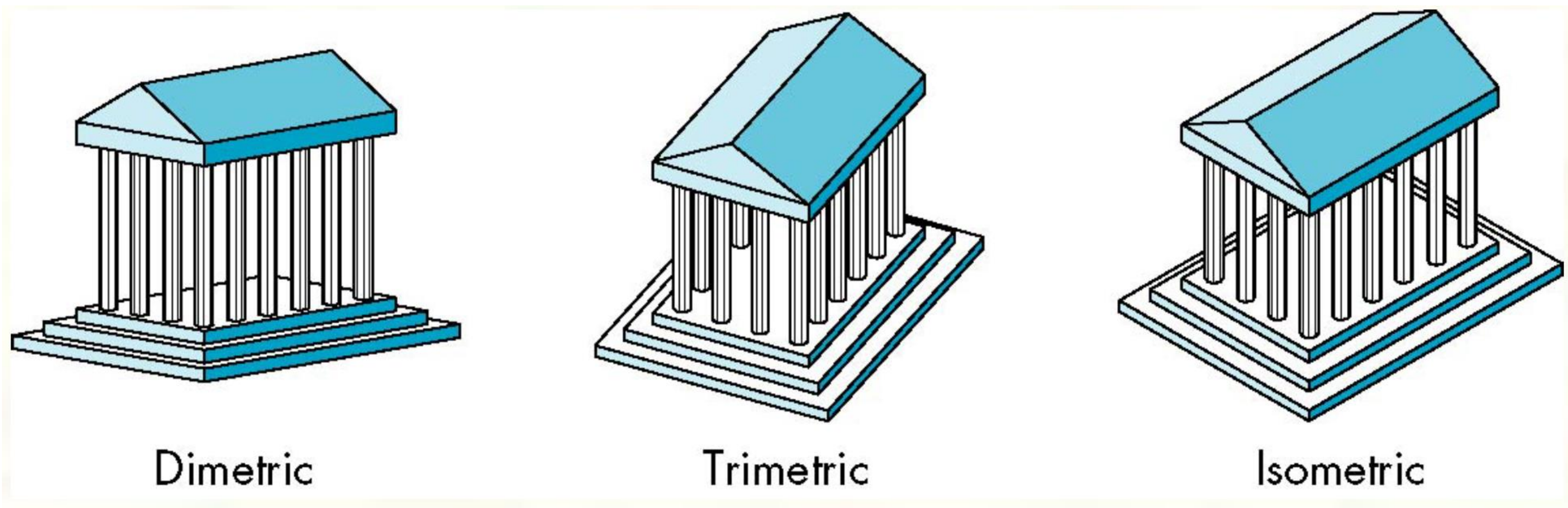
classify by how many angles of a corner of a projected cube are the same

none: trimetric
two: dimetric
three: isometric





Types of Axonometric Projections



Discussion

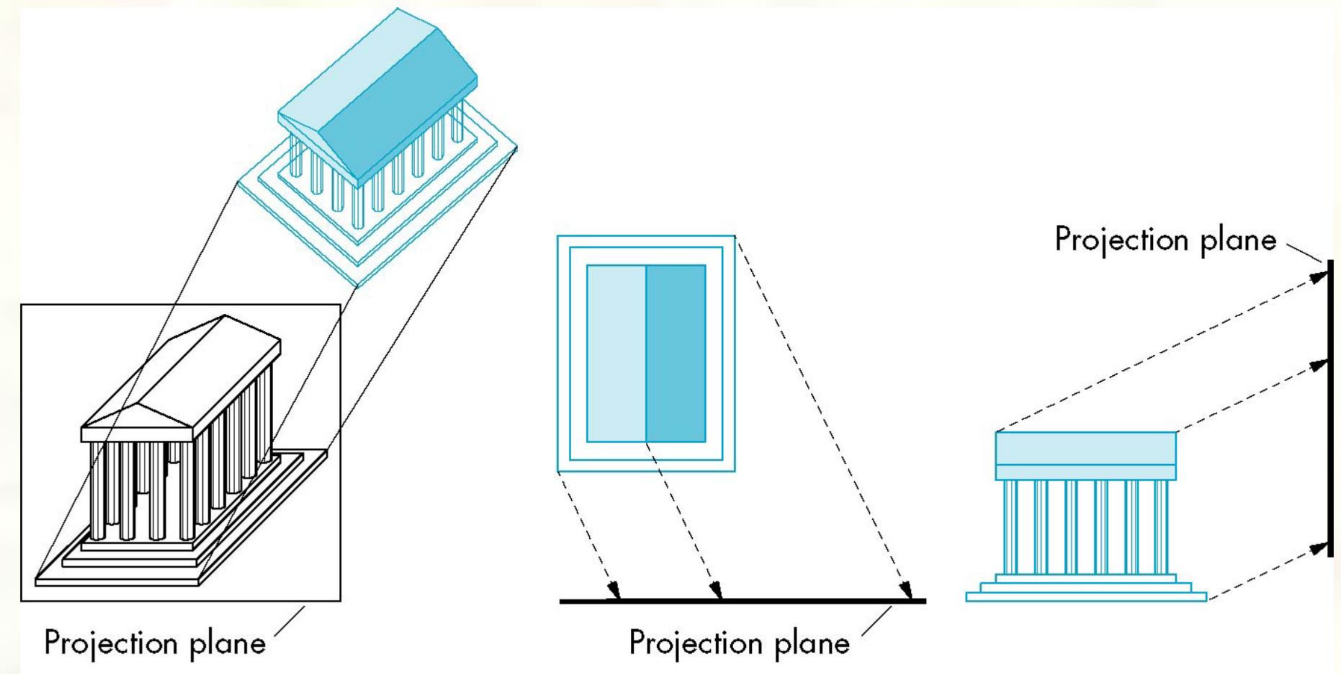
- Lines are scaled (*foreshortened*) but can find scaling factors
- Lines preserved but angles are not
  - Projection of a circle in a plane not parallel to the projection plane is an ellipse

- Can see three principal faces of a box-like object
- Some optical illusions possible
  - Parallel lines appear to diverge

- Does not look real because far objects are scaled the same as near objects
  - Used in CAD applications

Oblique Projection

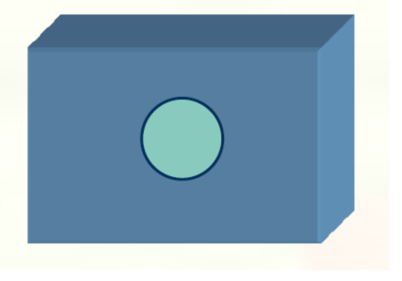
Arbitrary relationship between projectors and projection plane



The obverse of an orthographic projection is an oblique projection, which is a parallel projection in which the projection lines are not orthogonal to the projection plane.

Discussion

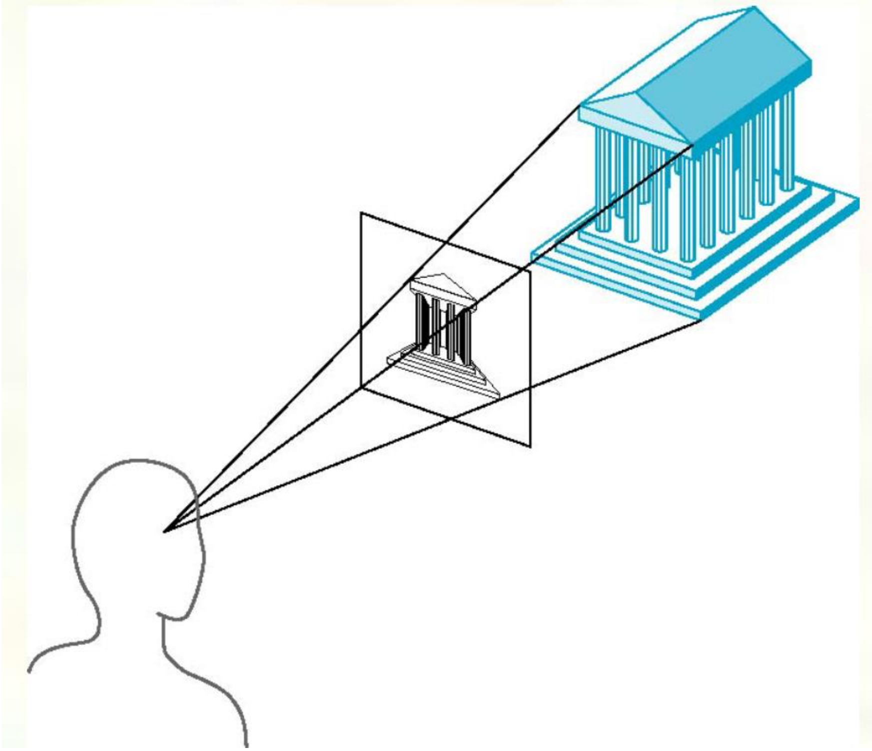
- Can pick the angles to emphasize a particular face
  – Architecture: plan oblique, elevation oblique
- Angles in faces parallel to projection plane are preserved while we can still see “around” side



- In physical world, cannot create with simple camera; possible with bellows camera or special lens (architectural)

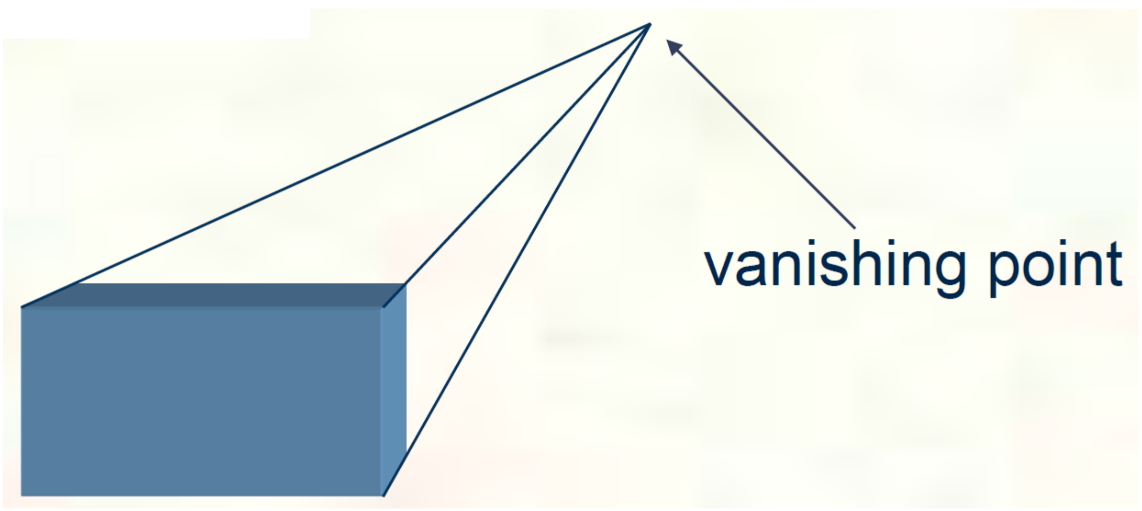
Perspective Projection

Projectors converge at center of projection



Vanishing Points

- Parallel lines (not parallel to the projection plan) on the object converge at a single point in the projection (the *vanishing point*)



Three-Point Perspective

- No principal face parallel to projection plane
- Three vanishing points for cube

Two-Point Perspective

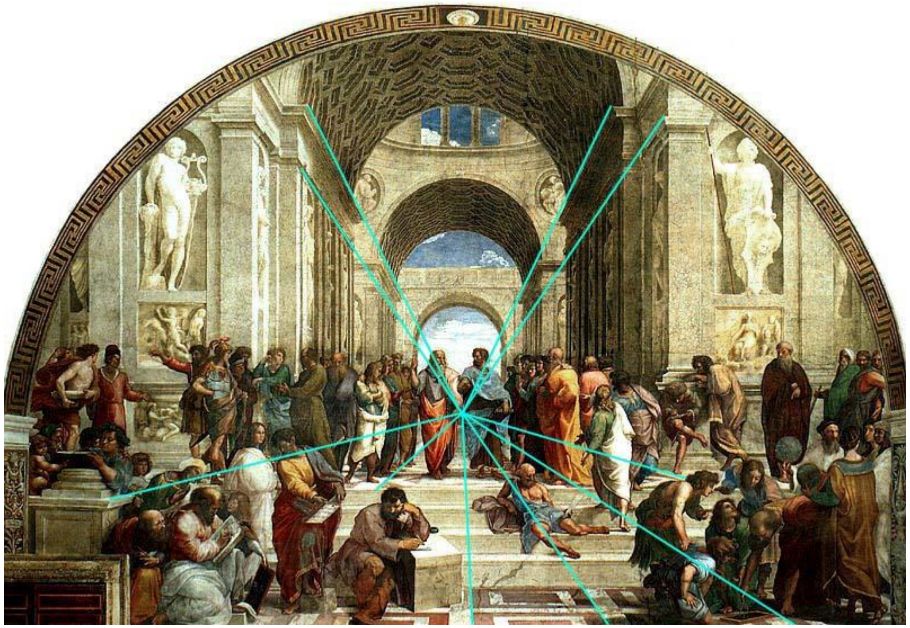
- On principal direction parallel to projection plane
- Two vanishing points for cube

One-Point Perspective

- One principal face parallel to projection plane
- One vanishing point for cube

Example

pre-renaissance often show poor understanding of perspective



Raphael's "The School of Athens" shows architectural perspecive to good effect

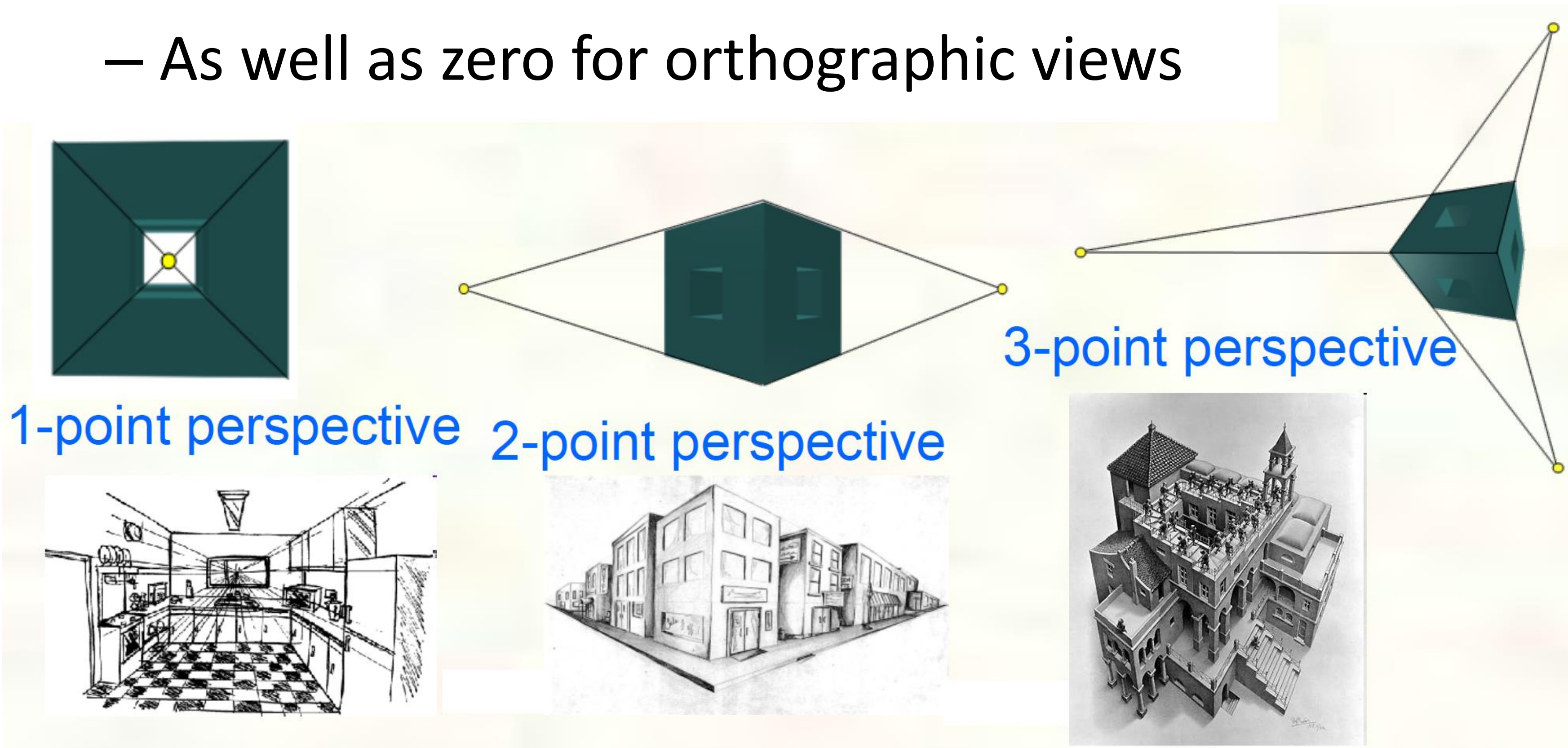
Image from https://en.wikipedia.org/wiki/File:Reconstruction_of_the_temple_of_Jerusalem.jpg
http://glasnost.itcarlow.ie/~powerk/GeneralGraphicsNotes/projection/perspective_projection.html

Discussion

- Objects further from viewer are projected smaller than the same sized objects closer to the viewer (*diminution*)
  - Looks realistic
- Equal distances along a line are not projected into equal distances (*nonuniform foreshortening*)
- Angles preserved only in planes parallel to the projection plane
- More difficult to construct by hand than parallel projections (but not more difficult by computer)

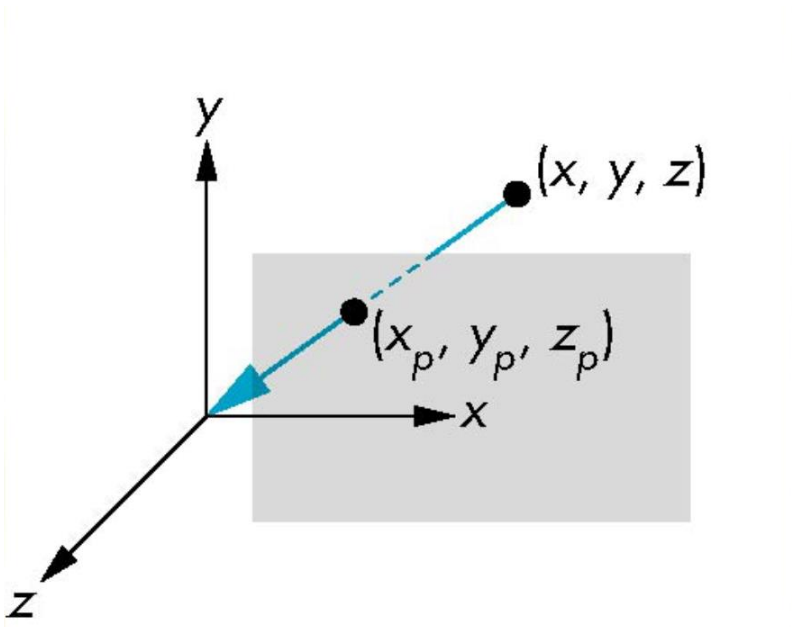
1-, 2-, and 3-point Perspective

- A 4x4 matrix can represent 1, 2, or 3 vanishing points
  - As well as zero for orthographic views



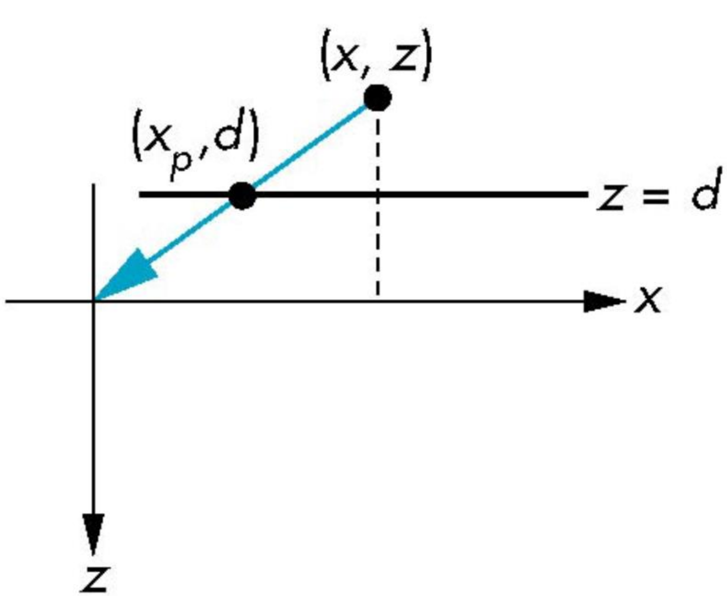
Simple Perspective

- Center of projection at the origin
- Projection plane $z = d$, $d < 0$

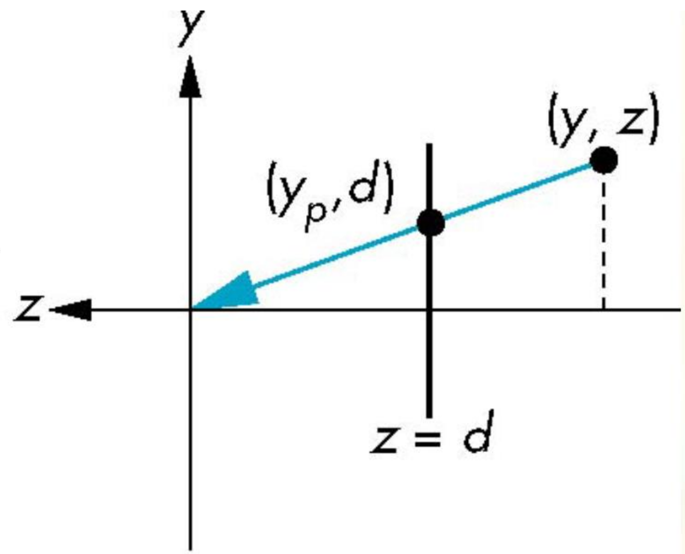


Perspective Equations

Consider top and side views



$$\frac{x}{z/d}$$



$$x_p = \frac{x}{z/d} \qquad y_p = \frac{y}{z/d} \qquad z_p = d$$

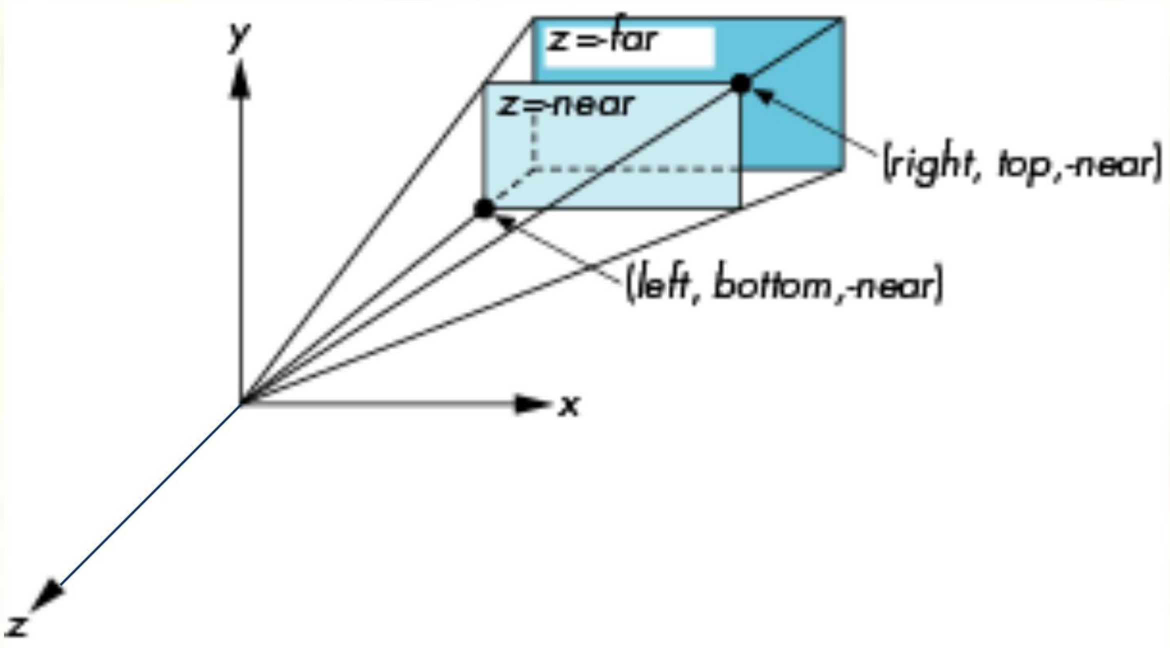
Homogeneous Form

consider $\mathbf{q} = \mathbf{Mp}$ where $\mathbf{M} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1/d & 0 \end{bmatrix}$

$$\mathbf{q} = \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \quad \Rightarrow \quad \mathbf{p} = \begin{bmatrix} x \\ y \\ z \\ z/d \end{bmatrix}$$

OpenGL Perspective

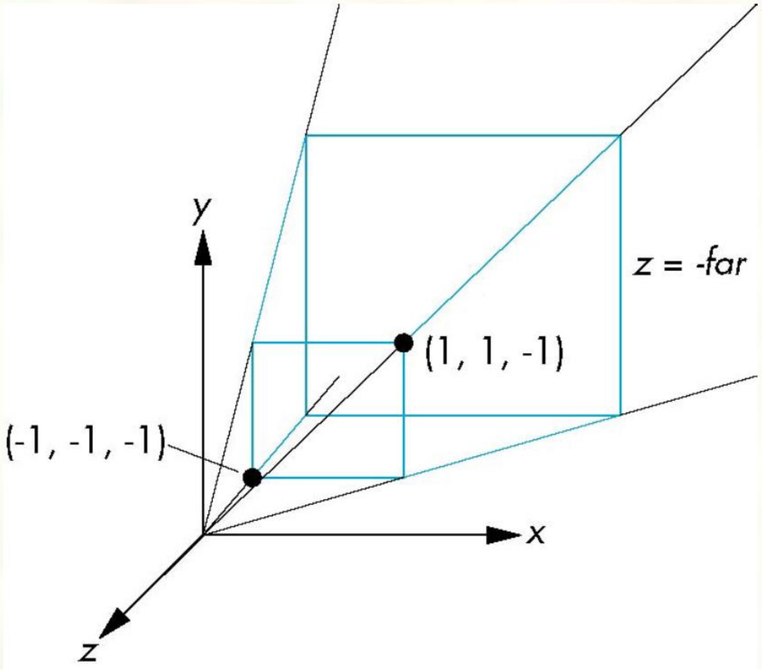
```
glFrustum(left,right,bottom,top,near,far)
```



Simple Perspective

Consider a simple perspective with the COP at the origin, the near clipping plane at z = -1, and a 90 degree field of view determined by the planes

$$x = \pm z, y = \pm z$$



Simple Eye to NDC

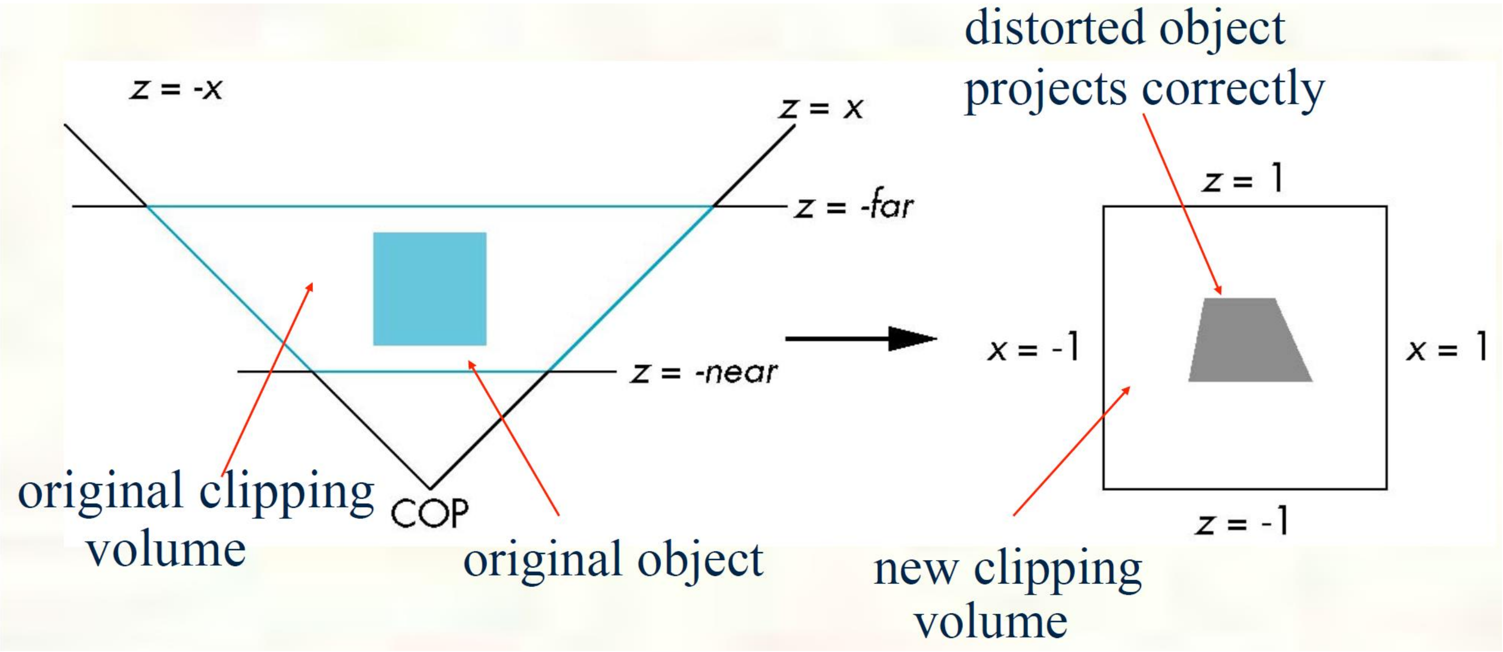
$$\mathbf{N} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \alpha & \beta \\ 0 & 0 & -1 & 0 \end{bmatrix}$$

after perspective division, the point $(x, y, z, 1)$ goes to

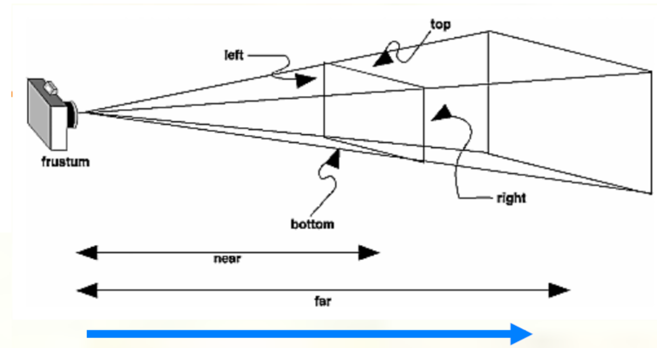
$x' = x/z$
$y' = y/z$
$z' = -(\alpha+\beta/z)$

which projects orthogonally to the desired point regardless of α and β

Normalization Transformation



glFrustum Example



-Z axis

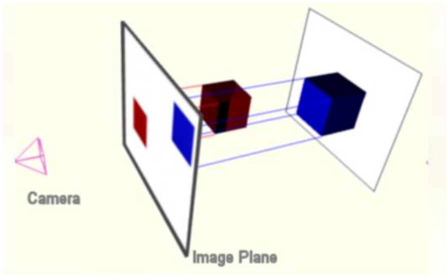
- Consider
  - glLoadIdentity();
    glFrustum(-30, 30, -20, 20, 1, 1000)
    - left=-30, right=30, bottom=-20, top=20, near=1, far=1000
- Matrix

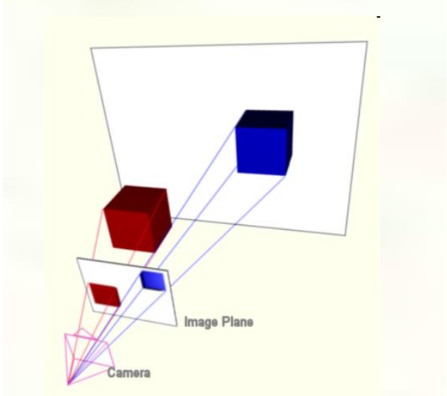
symmetric left/right & top/bottom so zero

$$\begin{bmatrix} \frac{2n}{r-l} & 0 & \frac{r+l}{r-l} & 0 \\ 0 & \frac{2n}{t-b} & \frac{t+b}{t-b} & 0 \\ 0 & 0 & \frac{-(f+n)}{f-n} & \frac{-2fn}{f-n} \\ 0 & 0 & -1 & 0 \end{bmatrix} = \begin{bmatrix} \frac{1}{30} & 0 & 0 & 0 \\ 0 & \frac{1}{20} & 0 & 0 \\ 0 & 0 & -\frac{1001}{999} & -\frac{2000}{999} \\ 0 & 0 & -1 & 0 \end{bmatrix}$$

glOrtho and glFrustum

- These OpenGL commands provide a parameterized transform mapping eye space into the “clip cube”
- Each command
  - glOrtho is orthographic
  - glFrustum is single-point perspective





Questions?