Projections

CSE 457, Autumn 2003

Graphics

http://www.cs.washington.edu/education/courses/457/03au/

Readings and References

Readings

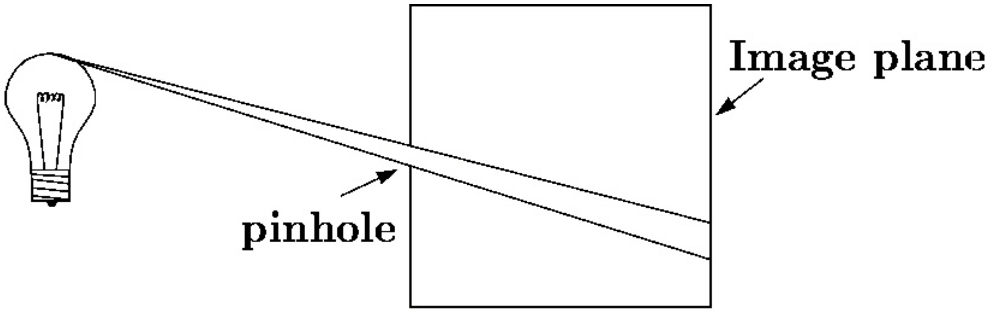
- Sections 5.2.2 – 5.2.4, *3D Computer Graphics*, Watt

Other References

- Chapter 5.6 and Chapter 6, *Computer Graphics*, *Principles and Practice*, Foley, van Dam

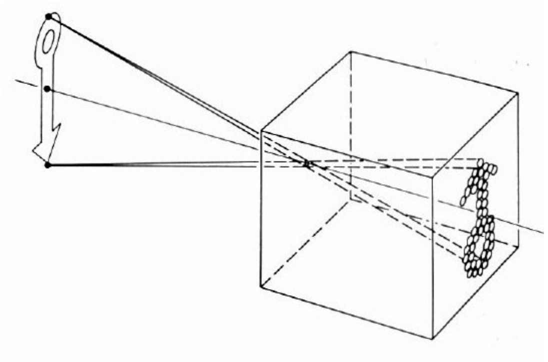
The pinhole camera

- The first camera - “camera obscura” - known to Aristotle.



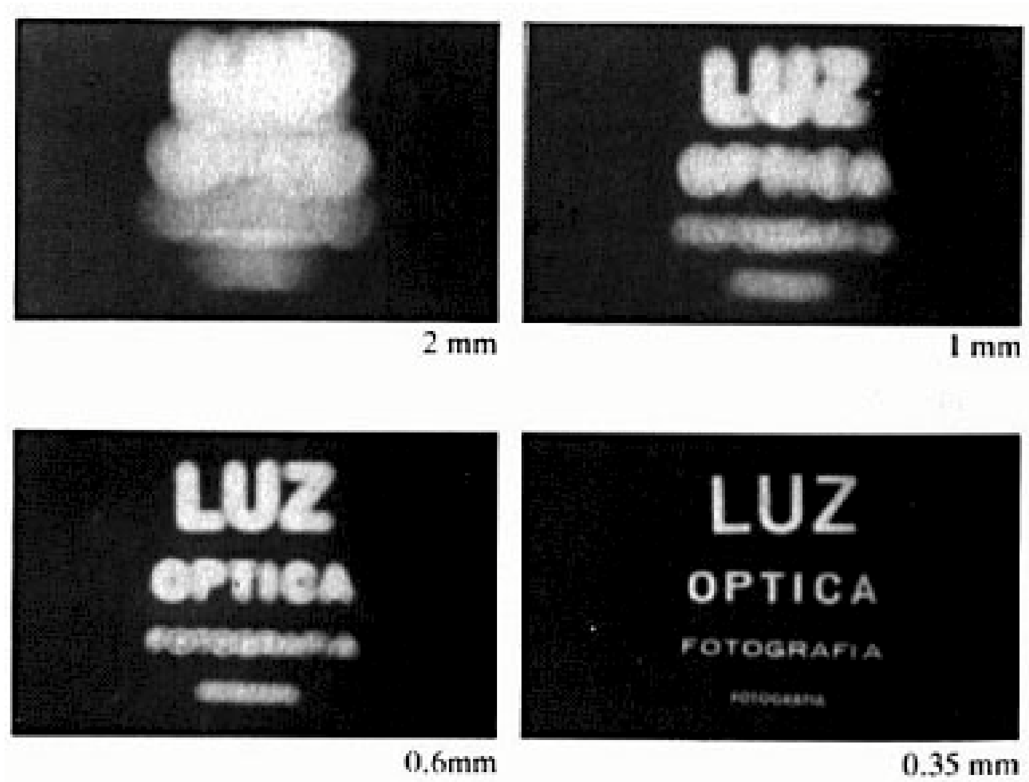
Pinhole camera

- In 3D, we can visualize the blur induced by the pinhole (a.k.a., **aperture**):

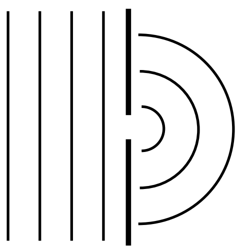


- **Q**: How would we reduce blur?

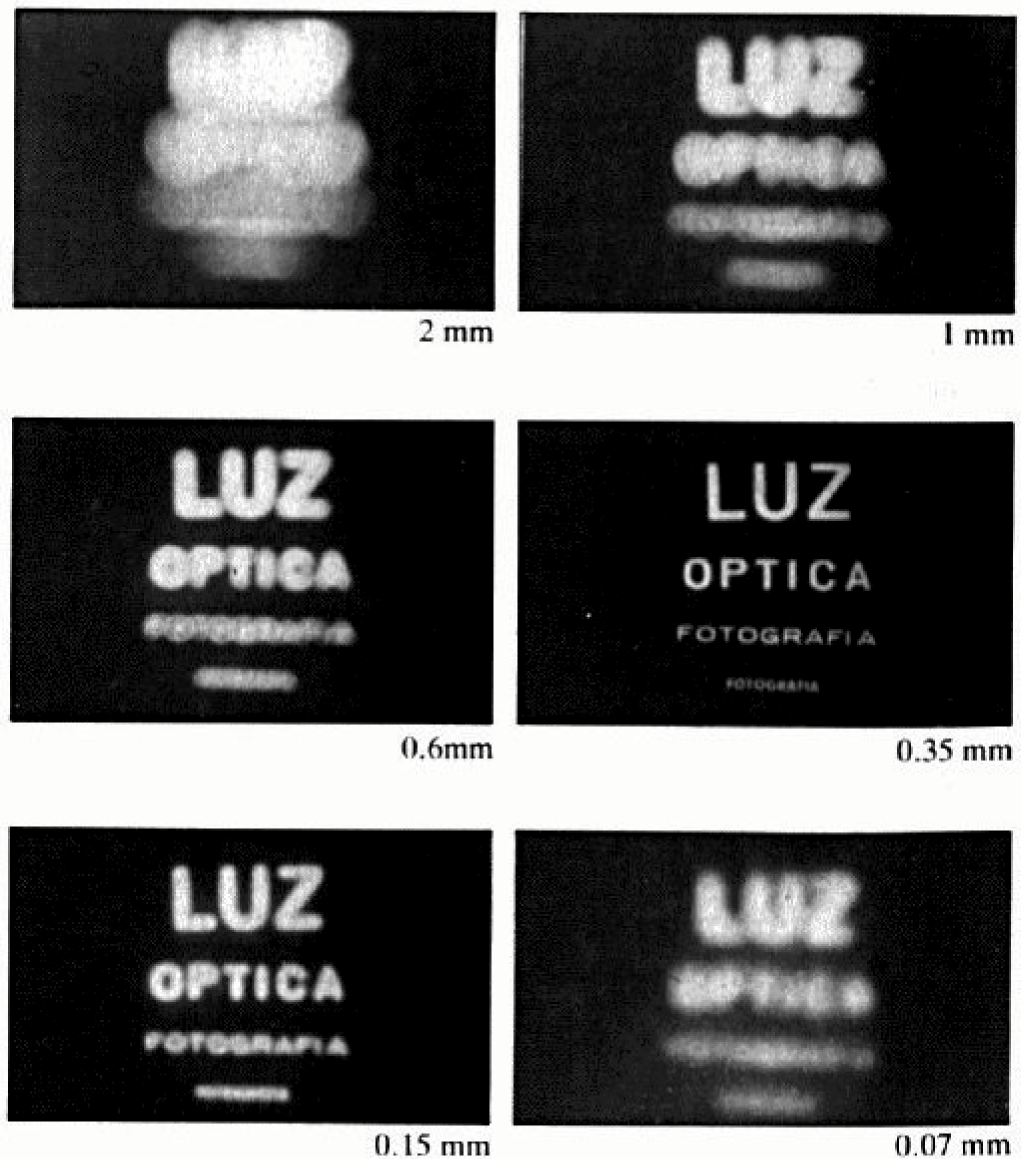
Shrinking the pinhole



- **Q**: What happens as we continue to shrink the aperture?



Diffraction



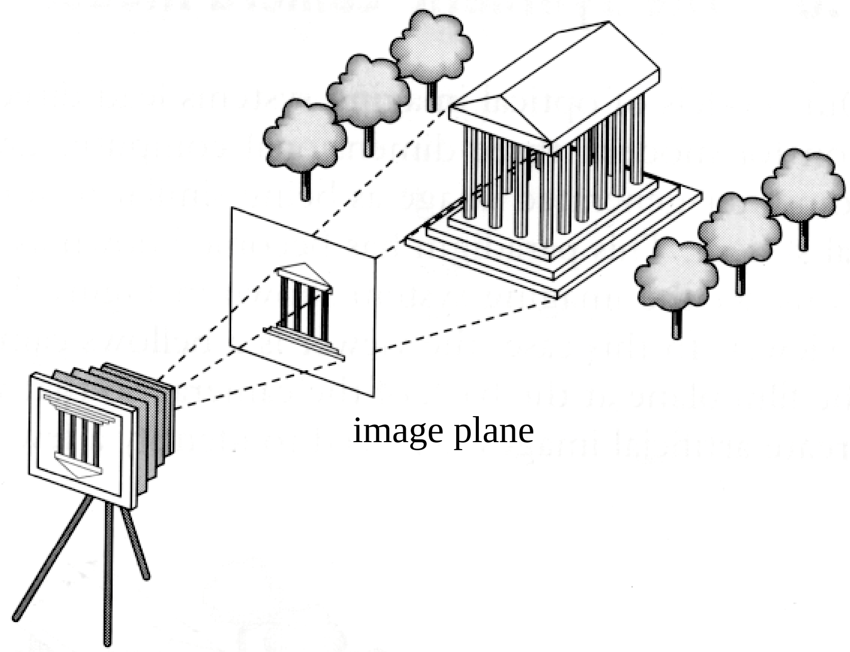
Imaging with the synthetic camera

- In practice, pinhole cameras require long exposures, can suffer from diffraction effects, and give an inverted image.
- In graphics, none of these physical limitations is a problem.

The image is rendered onto an **image plane** (usually in front of the "camera").

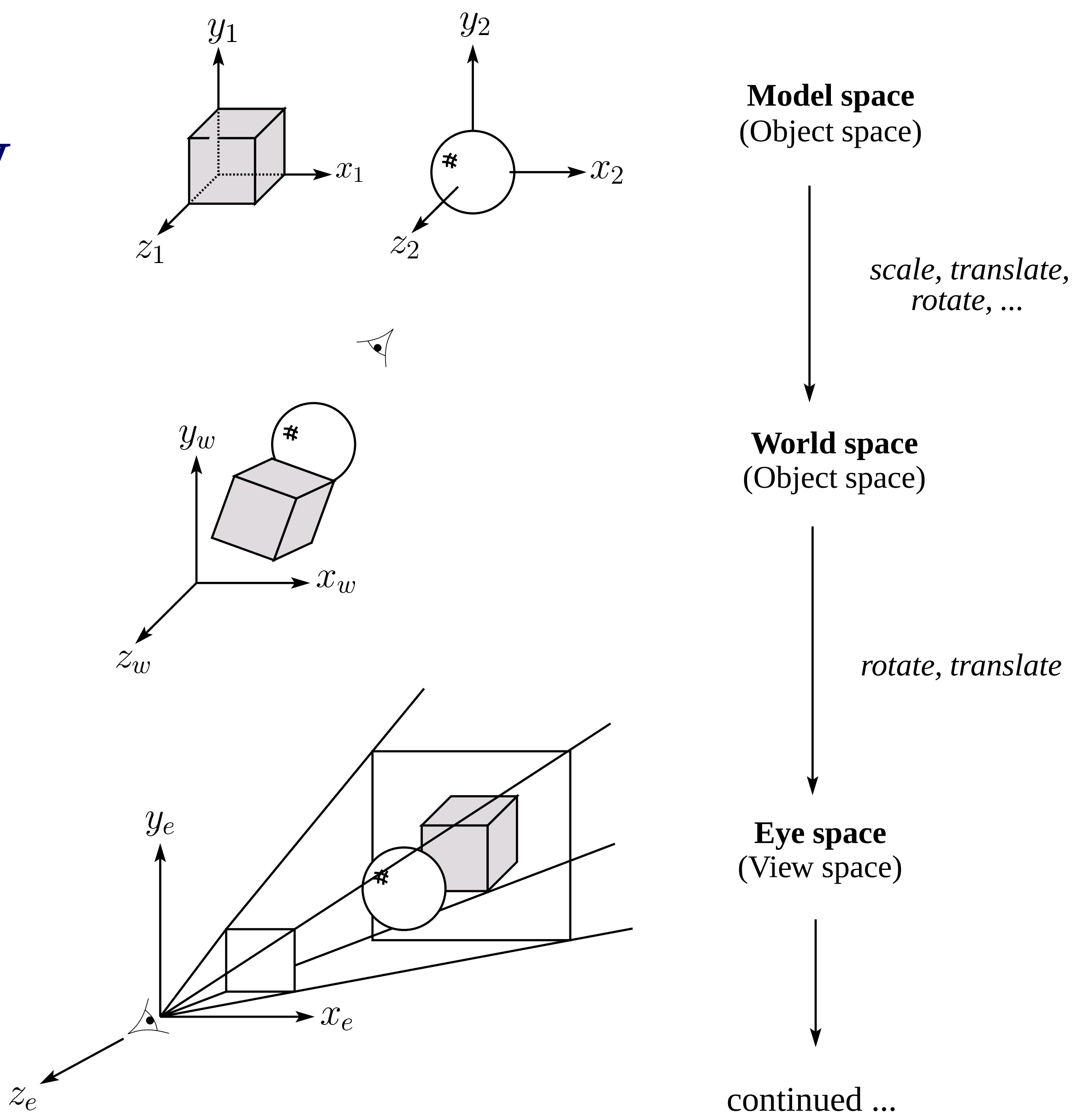
Viewing rays emanate from the **center of projection** (COP) at the center of the pinhole.

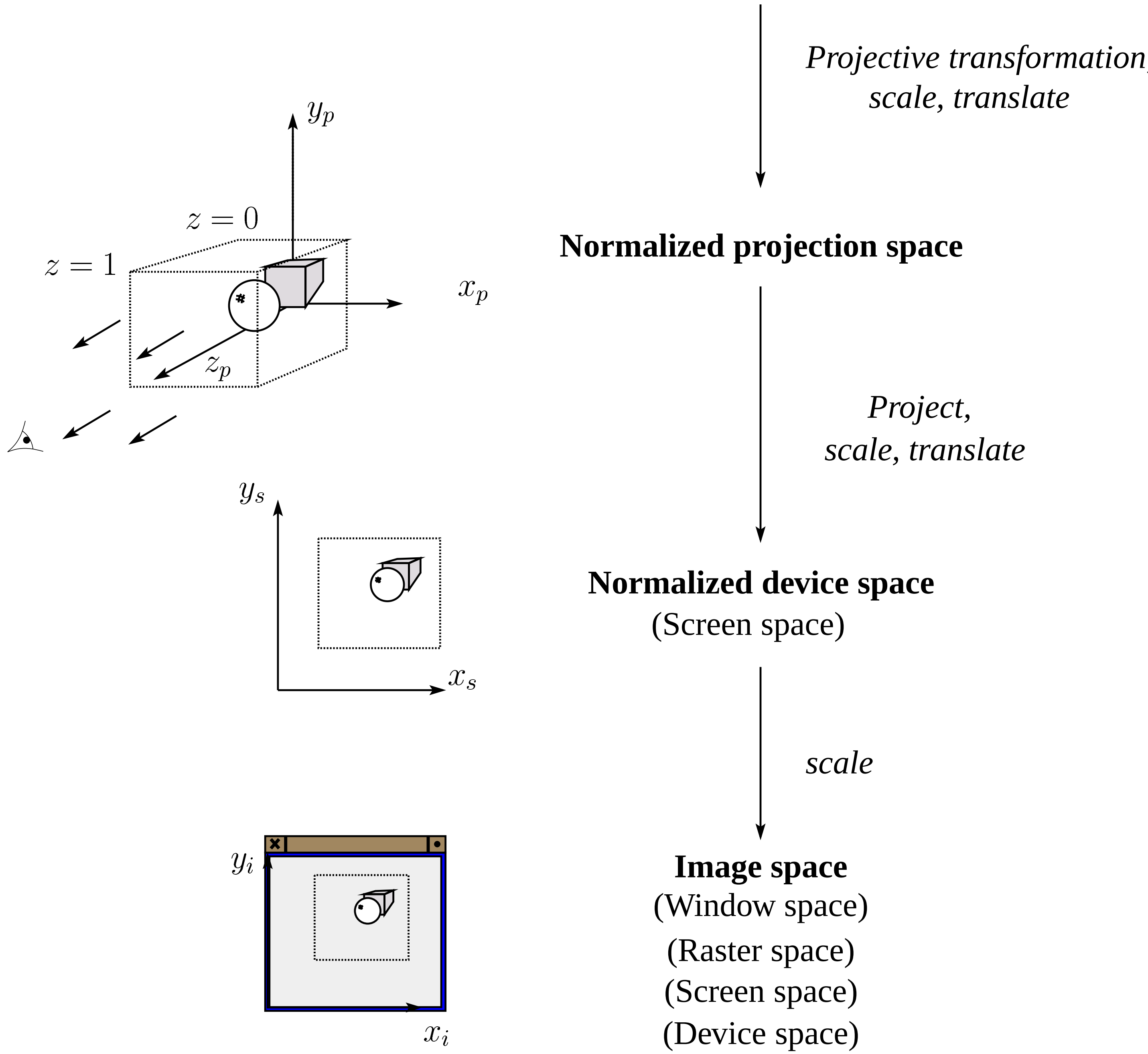
The image of an object point *P* is at the intersection of the viewing ray through *P* and the image plane.



3D Geometry Pipeline

Before being turned into pixels, a piece of geometry goes through a number of transformations...

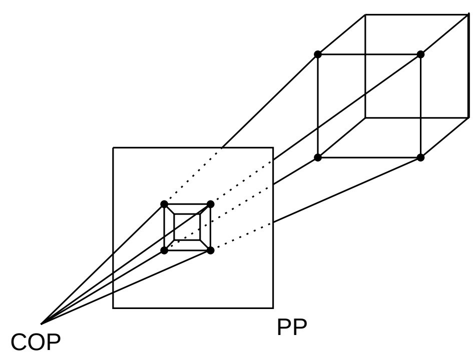


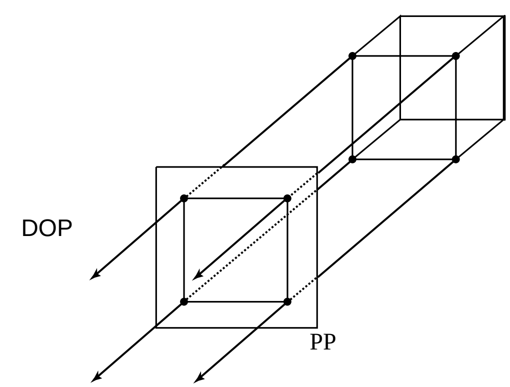


Projective transformation,
scale, translate
Normalized projection space
Project,
scale, translate
Normalized device space
(Screen space)
scale
Image space
(Window space)
(Raster space)
(Screen space)
(Device space)
y_p
$z = 0$
$z = 1$
x_p
z_p
y_s
x_s
y_i
x_i

Projections

- **Projections** transform points in *n*-space to *m*-space, where $m<n$.
- In 3-D, we map points from 3-space to the **projection plane** (PP) (a.k.a., image plane) along **projectors** (a.k.a., viewing rays) emanating from the center of projection (COP):





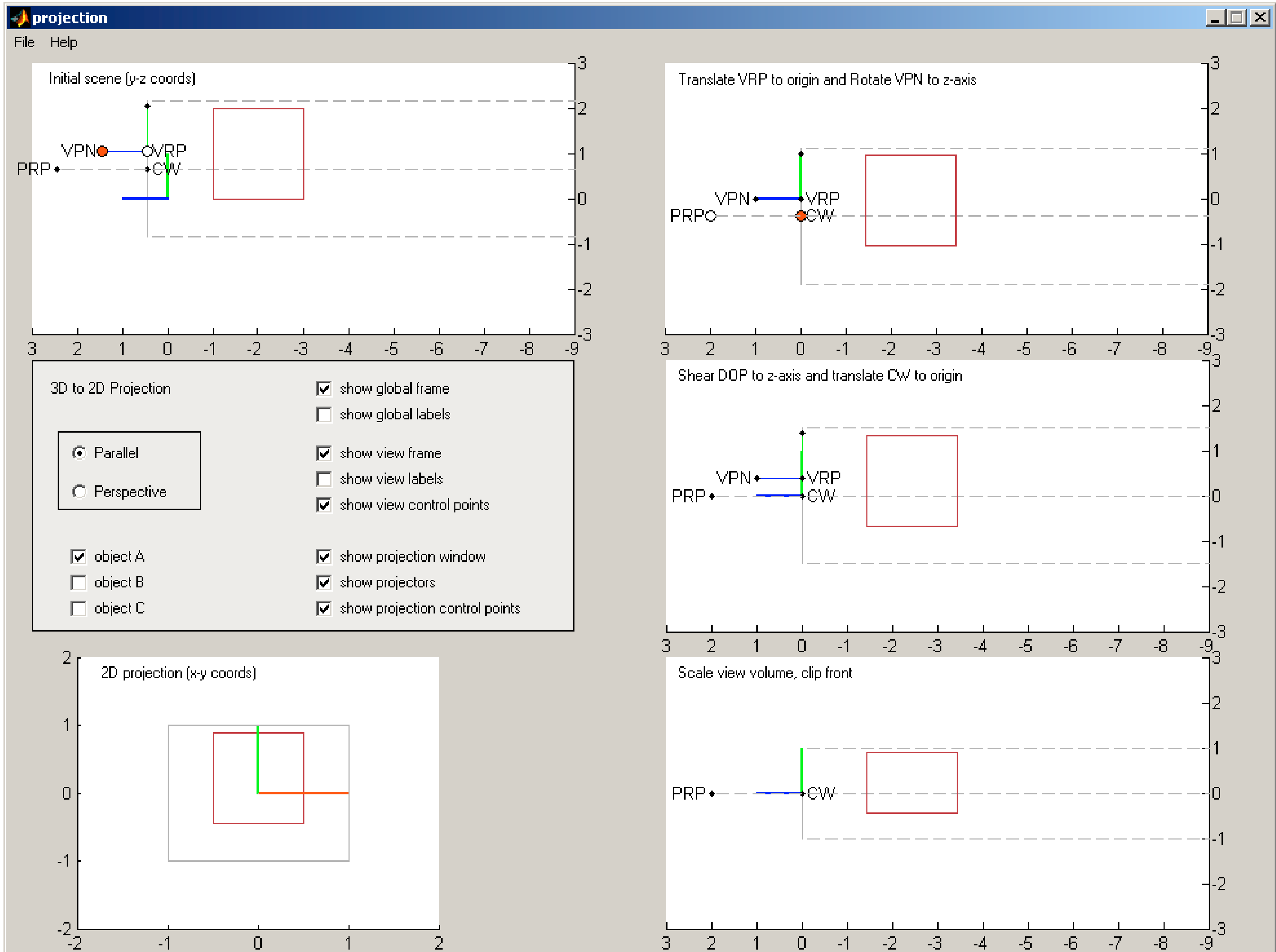
- There are two basic types of projections:
  - » Perspective – distance from COP to PP finite
  - » Parallel – distance from COP to PP infinite

Parallel projections

- For parallel projections, we specify a **direction of projection** (DOP) instead of a COP.
- There are two types of parallel projections:
  - » **Orthographic projection** – DOP perpendicular to PP
  - » **Oblique projection** – DOP not perpendicular to PP
- Orthographic projections along the z-axis in 3D or to 2D are easy

$$\begin{bmatrix} x' \\ y' \\ k \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & k \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \qquad \begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

  - » We can use shear to line things up when doing an oblique projection
  - » We often keep the initial z value around for later use. Why?


projection
File
Help
Initial scene (y-z coords)
VPN
VRP
PRP
CW
Translate VRP to origin and Rotate VPN to z-axis
VPN
VRP
PRP
CW
3D to 2D Projection
Parallel
Perspective
object A
object B
object C
show global frame
show global labels
show view frame
show view labels
show view control points
show projection window
show projectors
show projection control points
Shear DOP to z-axis and translate CW to origin
VPN
VRP
PRP
CW
2D projection (x-y coords)
Scale view volume, clip front
PRP
CW

Properties of parallel projection

- Properties of parallel projection:
  - » Not realistic looking
  - » Good for exact measurements
  - » Are actually a kind of affine transformation
    - Parallel lines remain parallel
    - Ratios are preserved
    - Angles not (in general) preserved
  - » Most often used in CAD, architectural drawings, etc., where taking exact measurement is important

Some oblique projections

Escaping Flatland is one of a series of sculptures by Edward Tufte

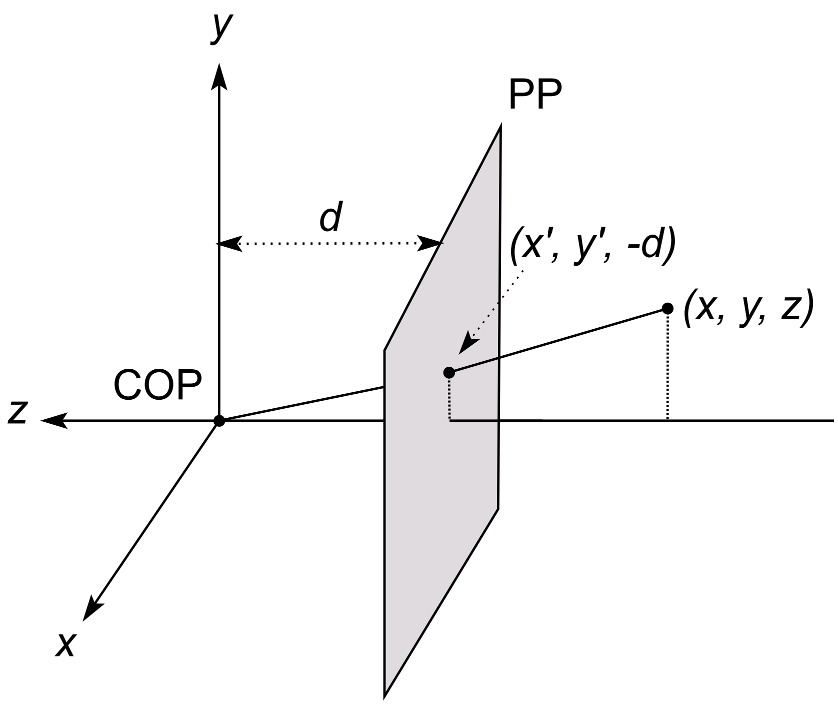
http://www.edwardtufte.com/tufte/sculpture

Larry Kagan:Wall Sculpture in Steel and Shadow

http://www.arts.rpi.edu/~kagan/

Derivation of perspective projection

- Consider the projection of a point onto the projection plane:



By similar triangles, we can compute how much the *x* and *y* coordinates are scaled:

Watt uses a left-handed coordinate system, and he looks down the +*z* axis, so his image plane is at +*d*.

Perspective projection

- The perspective projection as a matrix equation:

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{-1}{d} & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} x' \\ y' \\ w' \end{bmatrix} = \begin{bmatrix} x \\ y \\ \frac{-z}{d} \end{bmatrix}$$

- But remember we said that affine transformations work with the last coordinate always set to one.
  - » How can we bring this back to w'=1? Divide!
  - » This division step is the "perspective divide."

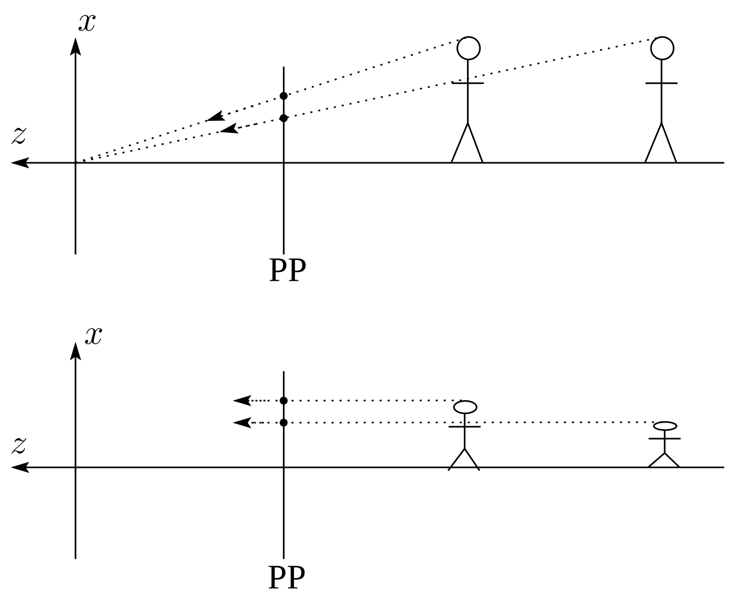
$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} -\frac{x}{z}d \\ -\frac{y}{z}d \\ 1 \end{bmatrix}$$

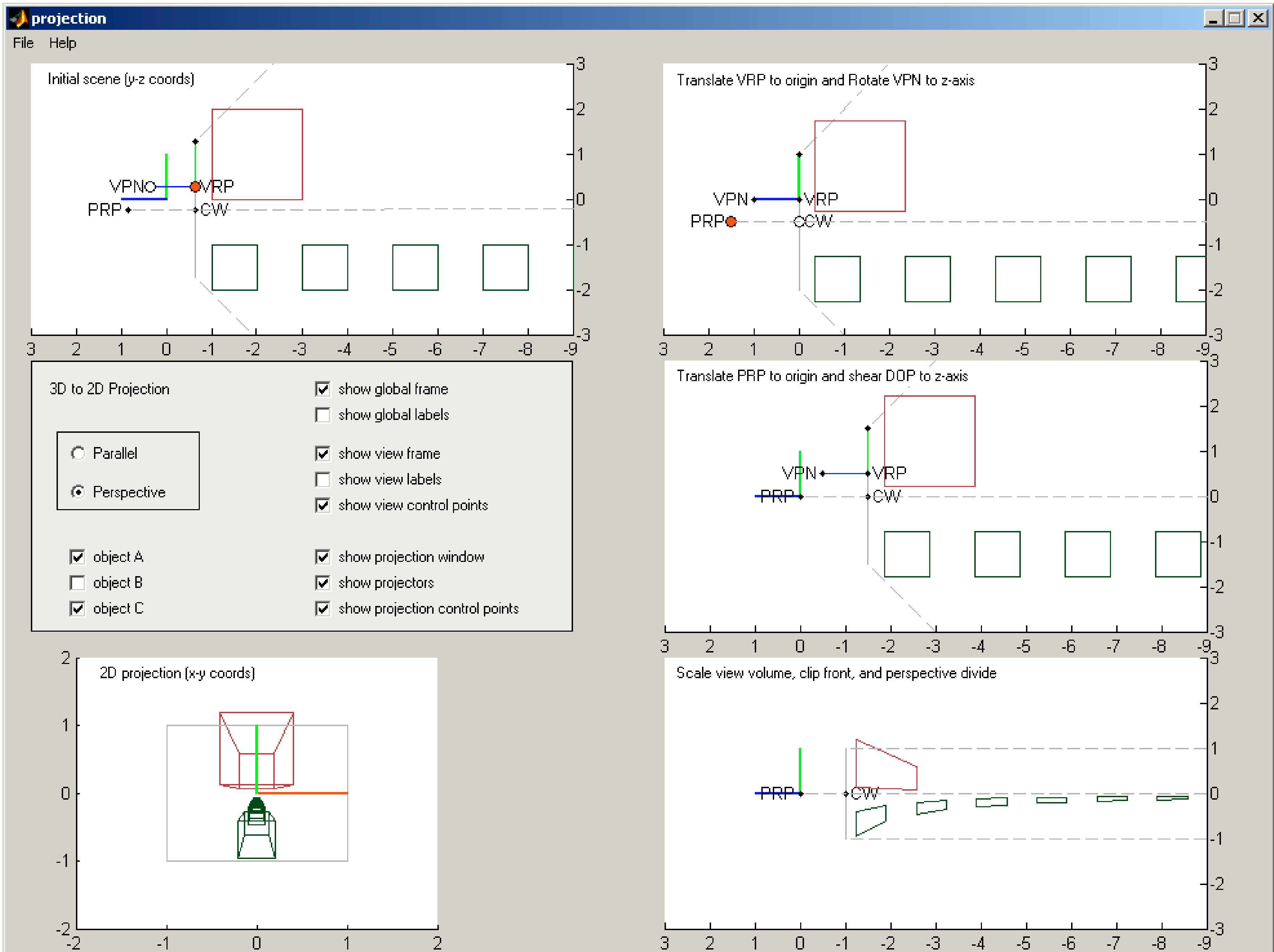
Again, projection implies dropping the *z* coordinate to give a 2D image, but we usually keep it around a little while longer.

Projective normalization

- After applying the perspective transformation and dividing by *w*, we are free to do a simple parallel projection to get the 2D image.

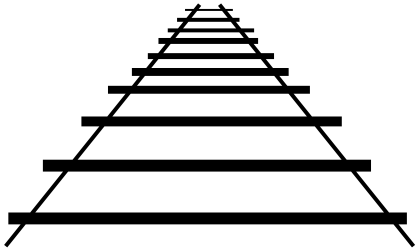
What does this imply about the shape of things after the perspective transformation + divide?




projection
File Help
Initial scene (y-z coords)
VPN
VRP
PRP
CW
Translate VRP to origin and Rotate VPN to z-axis
VPN
VRP
PRP
CW
3D to 2D Projection
Parallel
Perspective
object A
object B
object C
show global frame
show global labels
show view frame
show view labels
show view control points
show projection window
show projectors
show projection control points
Translate PRP to origin and shear DOP to z-axis
VPN
VRP
PRP
CW
2D projection (x-y coords)
Scale view volume, clip front, and perspective divide
PRP
CW

Vanishing points

- What happens to two parallel lines that are not parallel to the projection plane?
- Think of train tracks receding into the horizon...



- The equation for a line is:

$$\mathbf{l} = \mathbf{p} + t\mathbf{v} = \begin{bmatrix} p_x \\ p_y \\ p_z \\ 1 \end{bmatrix} + t \begin{bmatrix} v_x \\ v_y \\ v_z \\ 0 \end{bmatrix}$$

- After perspective transformation we get:

$$\begin{bmatrix} x' \\ y' \\ w' \end{bmatrix} = \begin{bmatrix} p_x + tv_x \\ p_y + tv_y \\ -(p_z + tv_z)/d \end{bmatrix}$$

Vanishing points (cont'd)

- Dividing by *w*:

$$\begin{bmatrix} x' \\ y' \\ w' \end{bmatrix} = \begin{bmatrix} -\dfrac{p_x + tv_x}{p_z + tv_z} d \\ -\dfrac{p_y + tv_y}{p_z + tv_z} d \\ 1 \\ \end{bmatrix}$$

- Letting *t* go to infinity:

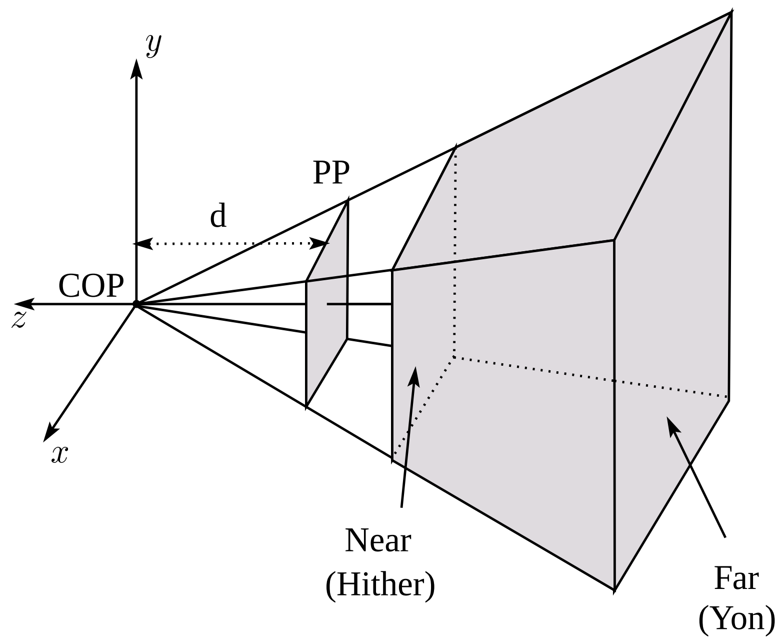
- We get a point!
- What happens to the line **l** = **q** + *t***v**?
- Each set of parallel lines intersect at a **vanishing point** on the Projection Plane.
- **Q**: How many vanishing points are there?

Clipping and the viewing frustum

- The center of projection and the portion of the projection plane that map to the final image form an infinite pyramid. The sides of the pyramid are **clipping planes**.

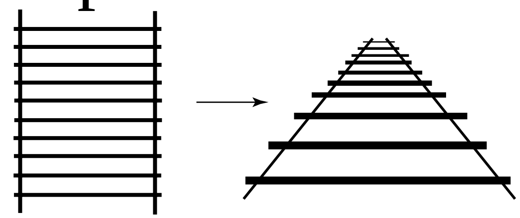
Frequently, additional clipping planes are inserted to restrict the range of depths. These clipping planes are called the **near** and **far** or the **hither** and **yon** clipping planes.

All of the clipping planes bound the **viewing frustum**.



Properties of perspective projections

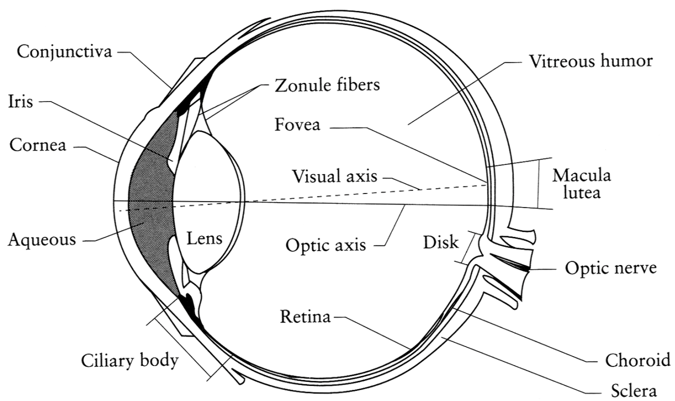
- The perspective projection is an example of a **projective transformation**.



- Some properties of projective transformations:
  - » Lines map to lines
  - » Parallel lines do <u>not</u> necessarily remain parallel
  - » Ratios are <u>not</u> preserved
- An advantage of perspective projection is that size varies inversely with distance – looks realistic.
- A disadvantage is that we can't judge distances as exactly as we can with parallel projections.

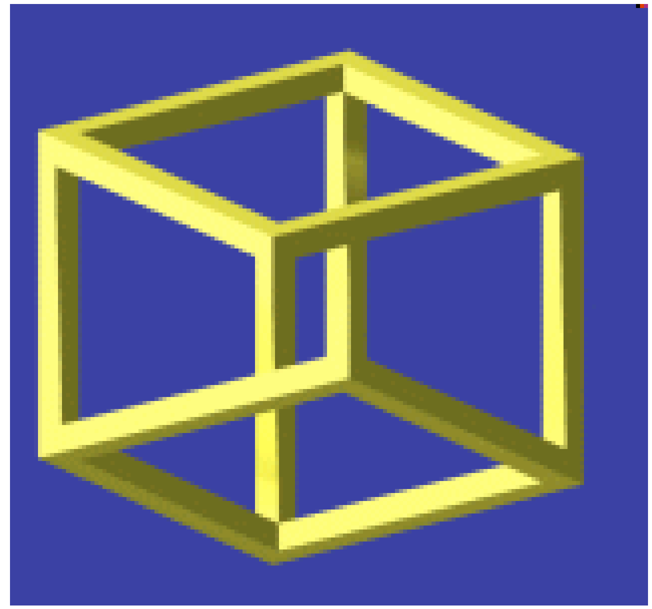
Human vision and perspective

- The human visual system uses a lens to collect light more efficiently, but records perspectively projected images much like a pinhole camera.



- **Q**: Why did nature give us eyes that perform perspective projections?
  - » How would you construct a vision system that did parallel projections?
- **Q**: Do our eyes "see in 3D"?

Some View Transformations



http://www.ntv.co.jp/kasoh/past_movie/contents.html

http://www.cs.technion.ac.il/~gershon/EscherForReal/

Summary

- What to take away from this lecture:
  - » All the boldfaced words.
  - » An appreciation for the various coordinate systems used in computer graphics.
  - » How the perspective transformation works.
  - » How we use homogeneous coordinates to represent perspective projections.
  - » The classification of different types of projections.
  - » The concept of vanishing points
  - » The mathematical properties of projective transformations.