Projections

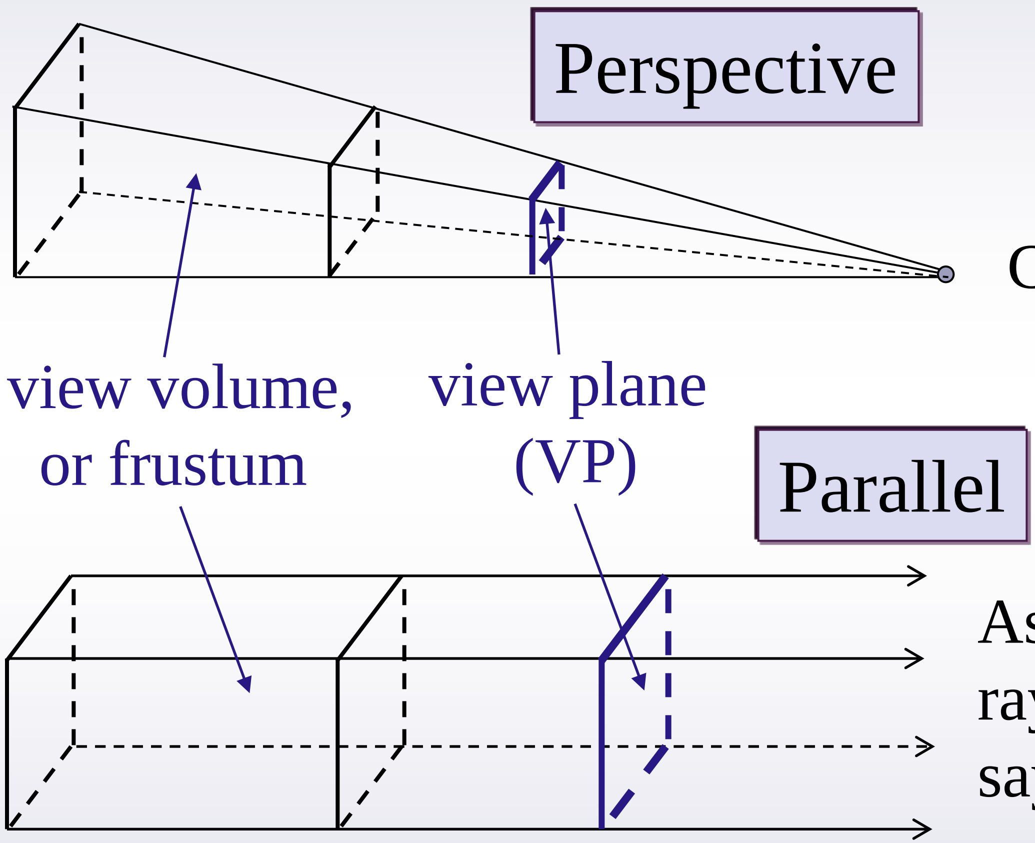
A projection is

- A mapping from $R^n \Rightarrow R^n$
  - we often assume z=0, but we are still in 3D
- Idempotent => P*P*…*P = P
  - subsequent projections have no effect

Two main types we will use in CG

- Parallel
- Perspective
- Can be combined for a generalized projection

Projections



COP (eye, origin of camera frame)

As COP moves to infinity, rays become parallel & we say DOP (direction of proj.)

Both perspective and parallel projections are *planar geometric projections* because the surface is a plane and the projectors are lines*

Perspective projections

Characteristics

- Parallel lines of the object that are not parallel to view plane converge to a vanishing point
- Closer objects look larger than farther objects
- Natural view, used in rendering and animation
- Does not preserve lengths or angles

Types

- One, two, three point perspectives
  - defined by number of principal axes cut by proj.

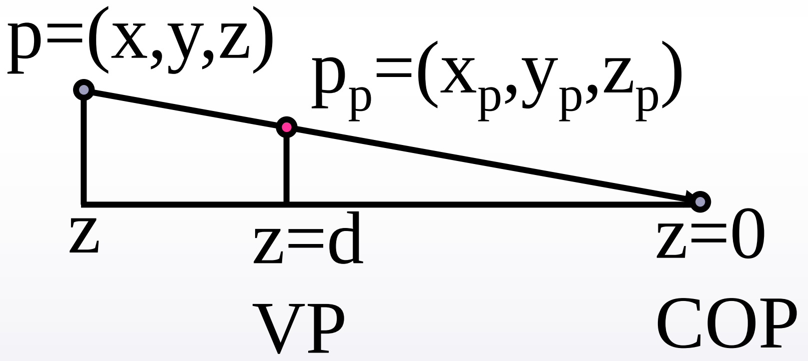
Perspective projections

- One-point
  - One principle axis cut by projection plane
  - One axis vanishing point
- Two-point
  - Two principle axes cut by projection plane
  - Two axis vanishing points
- Three-point
  - Three principle axes cut by projection plane
  - Three axis vanishing points

Perspective projections

Assume that the view plane (VP) is orthogonal to the z axis at z = d.

Find the projection p of a point p:



*d is a scale factor applied to x and y,
*division by z causes distant objects to appear smaller than closer objects
*perspective proj is irreversible (do you see why?)

By similar triangles,

$x_p/d = x/z \Rightarrow x_p = (d*x)/z = x/(z/d)$

$y_p/d = y/z \Rightarrow y_p = (d*y)/z = y/(z/d)$

Perspective projections

The fact that many points map to one point is a problem

- We need depth info for hidden surface removal
- Homogeneous coordinates allow a fix
- Use p = (x, y, z, w) instead of p = (x, y, z, 1)

$$\begin{bmatrix} x_p \\ y_p \\ z_p \\ 1 \end{bmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1/d & 0 \end{pmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} x \\ y \\ z \\ z/d \end{bmatrix} = \begin{pmatrix} x/(z/d) \\ y/(z/d) \\ d \\ 1 \end{pmatrix}$$

Note: w = z/d cannot be zero (=> pts on plane z=0 do not project)
(This division is not technically part of the projection.)

Generalized projections

$$M_{parallel} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Assumes:
DOP parallel to z axis

$$M_{perspective} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1/d & 0 \end{pmatrix}$$

Assumes:
COP at origin

A more robust formulation not only removes these restrictions but also integrates perspective and parallel projections into a single matrix

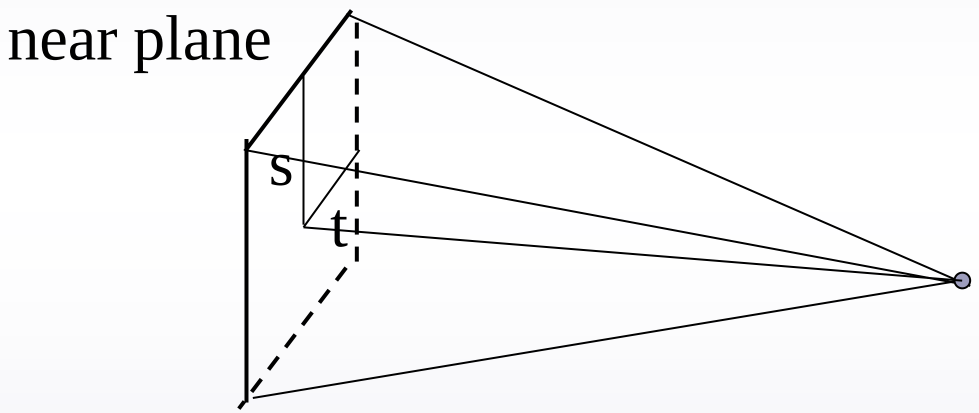
View volumes

Two types

- *View volume* for orthographic projection is a right parallelpiped
- *View frustum* for perspective is a clipped pyramid
- Defined by six clip planes, left/right, top/bottom, near/far (also hither/yon & front/back)

Perspective view frustum

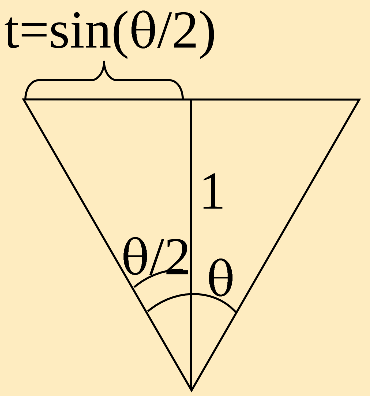
Define frustum by field of view and near and far clip planes



Find x extents (t):

fov = θ

left = -t, right = +t

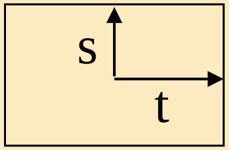


Find y extents (s):

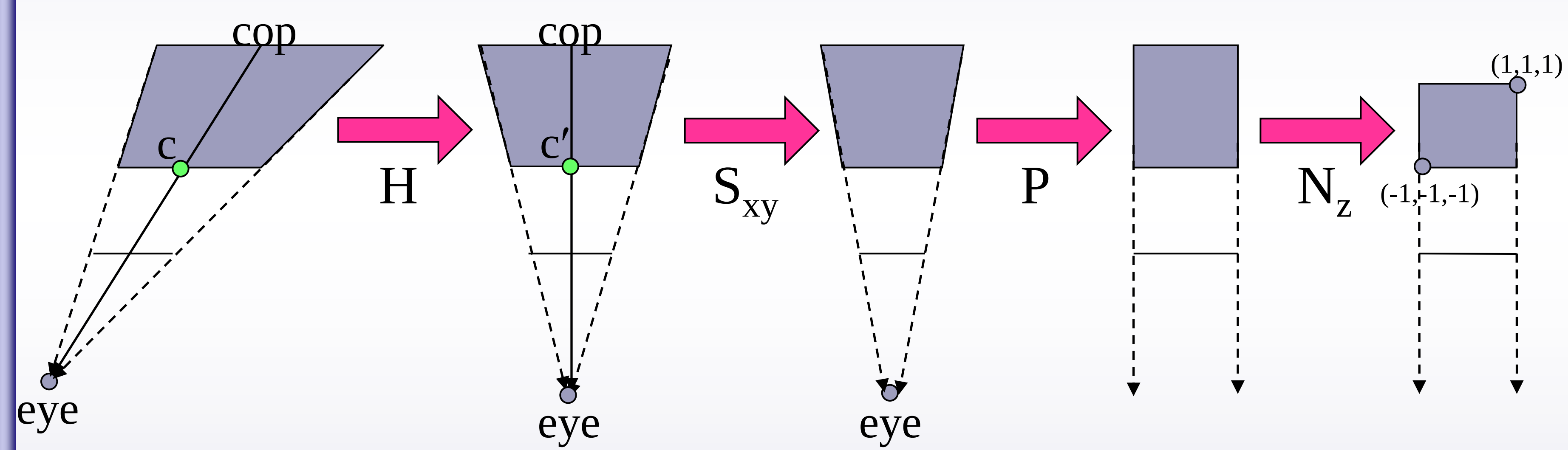
height/width = s/t

(keep aspect ratio of window)

=> s = t * height/width



General perspective projection



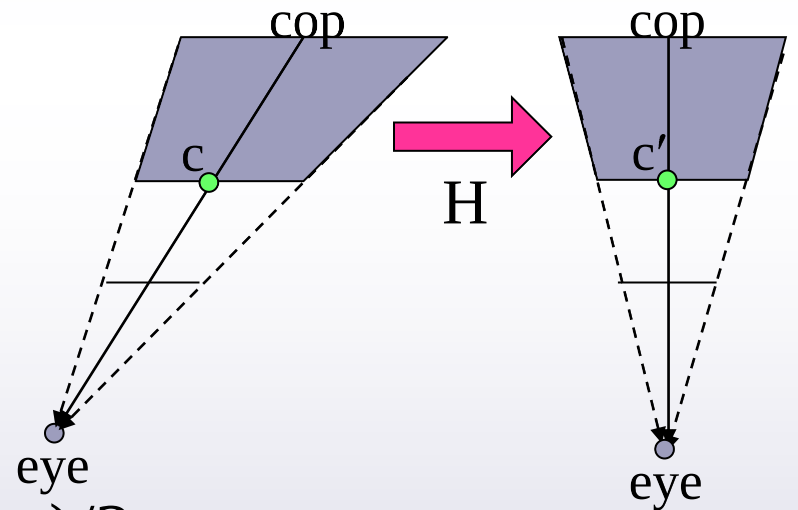
$$P = M_{orth} NPSH$$

General perspective proj.

1. Shear view volume so centerline of proj is perp to VP (shear in x,y along z)

shear c = ((l+r)/2, (t+b)/2, n) to c′ = (0,0,n)

$$H_{xy} = \begin{pmatrix} 1 & 0 & k_{xz} & 0 \\ 0 & 1 & k_{yz} & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$



where:

$k_{xz} = (l+r)/2n$

$k_{yz} = (t+b)/2n$

so that:

$x' = x + z(l+r)/2n$

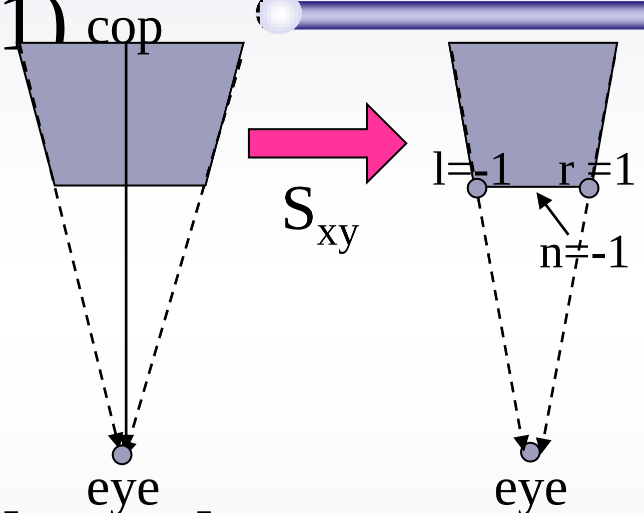
$y' = y + z(t+b)/2n$

$z' = z$

General perspective proj.

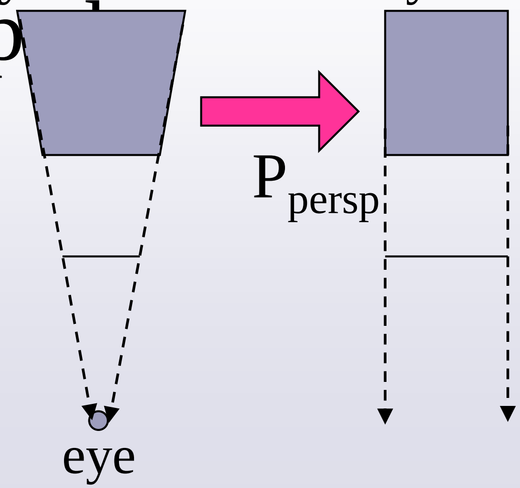
2. Scale sides of frustum, i.e. normalize in x,y (want l = -1, r = 1, n = -1)

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



3. Persp proj creates rect. parallelpip

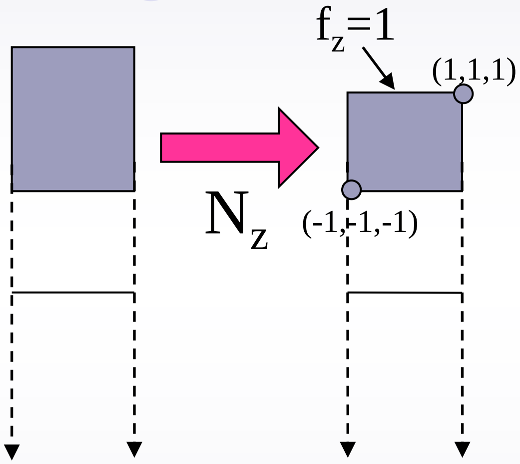
$$P_{persp.} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1/d & 0 \end{pmatrix}$$



General perspective proj.

4. Normalize in z
(scale and translate far plane)

$$N_z = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & (f+n)/(f-n) & 2fn/(f-n) \\ 0 & 0 & 1 & 1 \end{pmatrix}$$

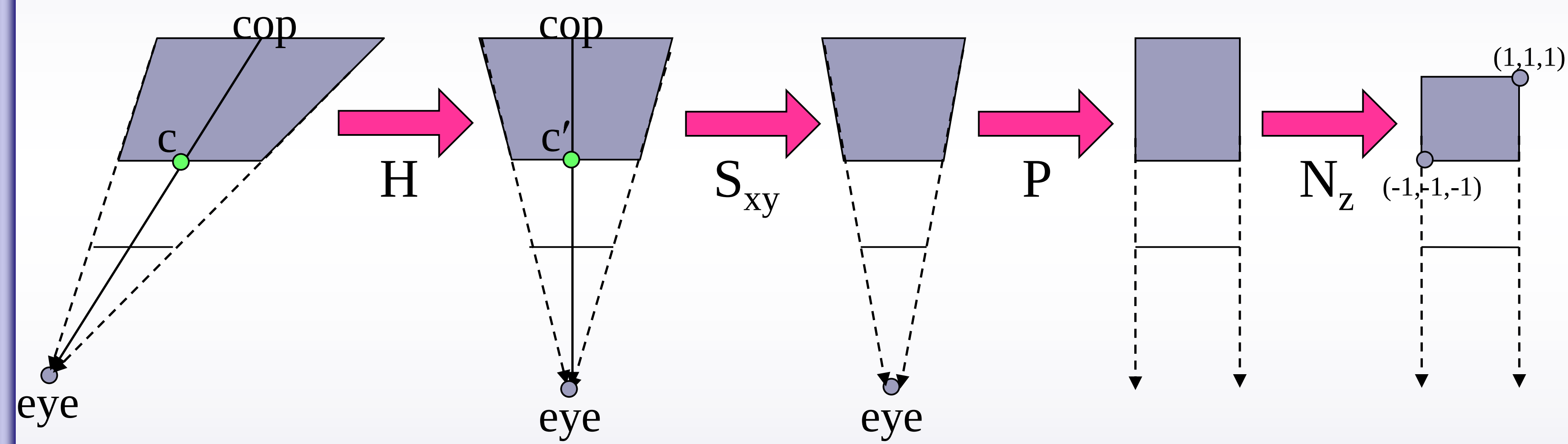


Final projection matrix is:

$$P = H_{xy}\, S_{xy}\, P_{persp}\, N_z$$

*in text, N includes P_{persp}

General perspective projection



$$P = M_{orth} NPSTH$$