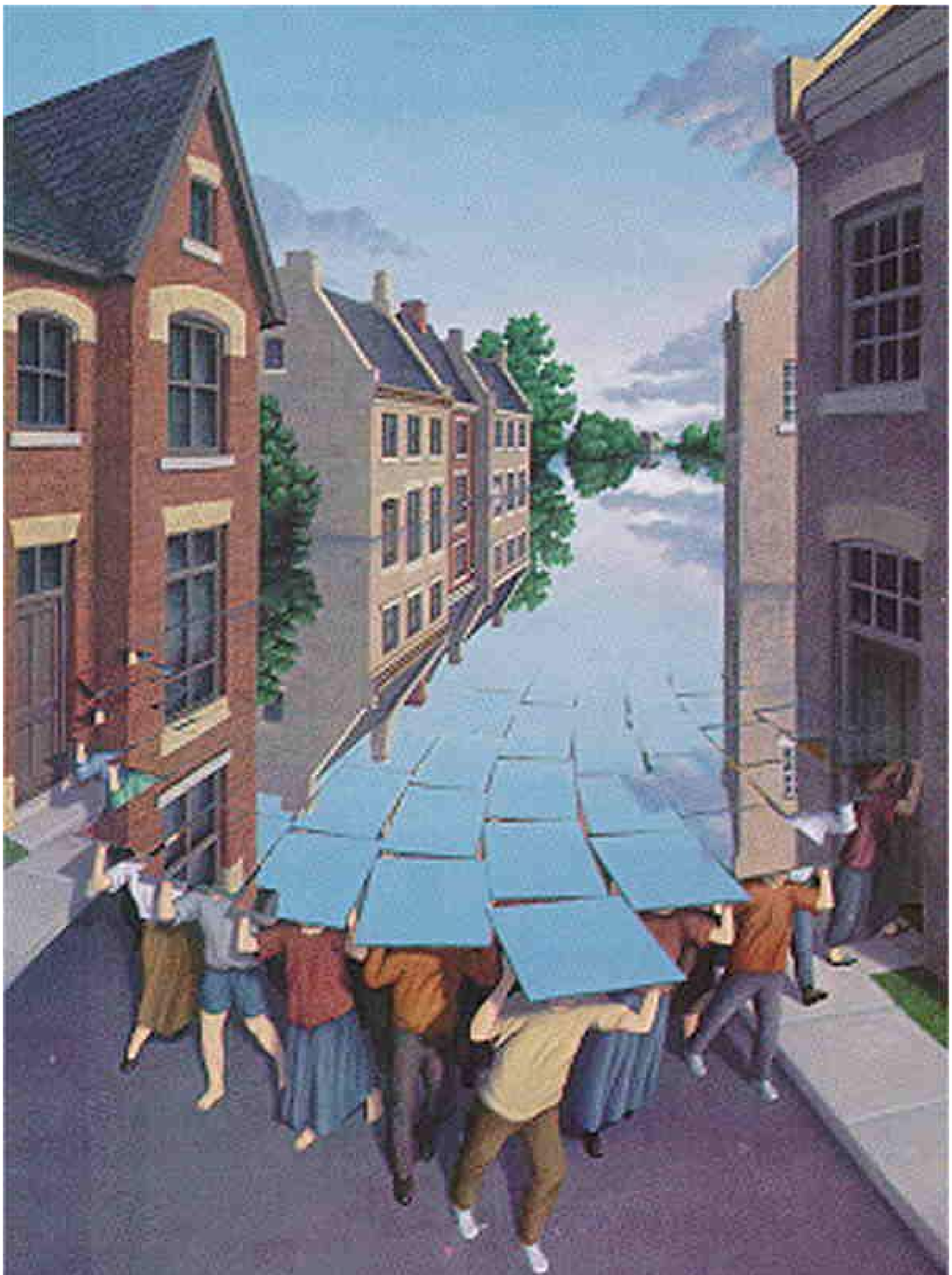
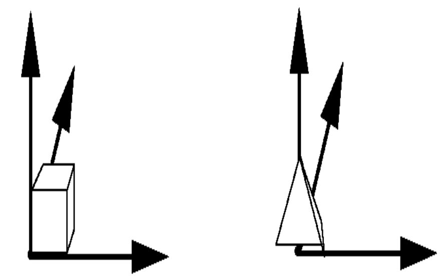
Viewing in 3D

(Chapt. 6 in FVD, Chapt. 12 in Hearn & Baker)

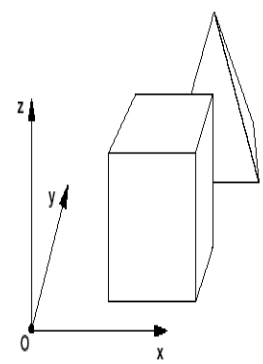


Common Coordinate Systems

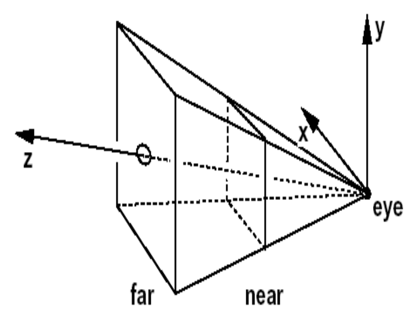
- Object space
  - local to each object



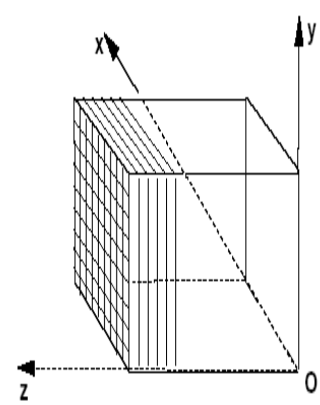
- World space
  - common to all objects



- Eye space / Camera space
  - derived from view frustum



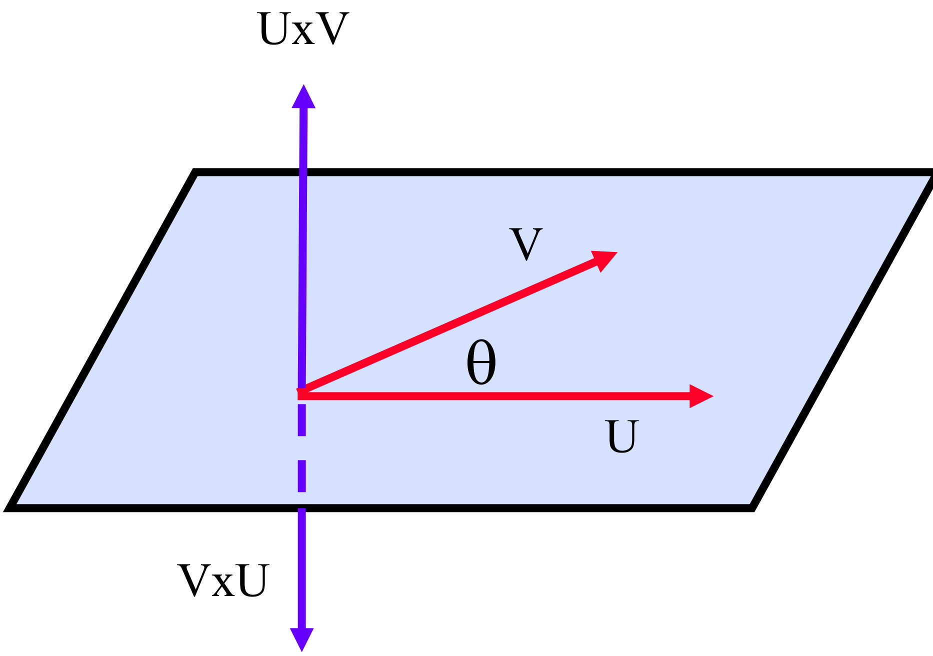
- Screen space
  - indexed according to hardware attributes



Specifying the Viewing Coordinates

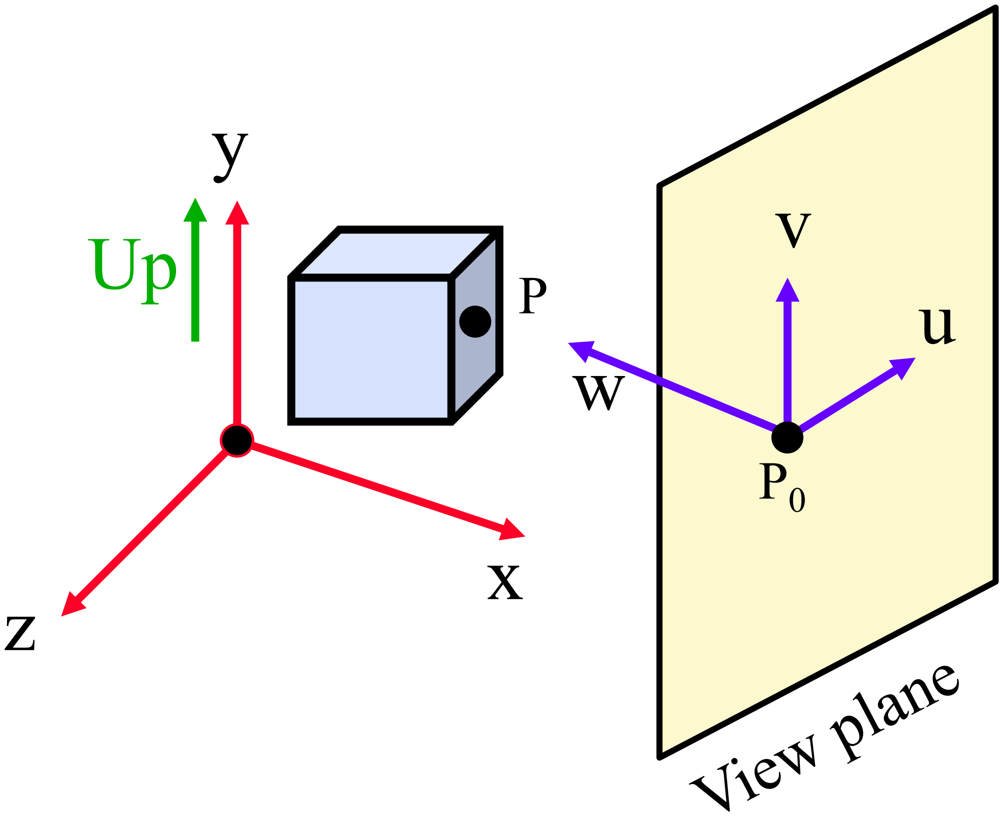
- *Viewing Coordinates system*, [u, v, w], describes 3D objects with respect to a viewer.
- A *viewing plane* (*projection plane*) is set up perpendicular to w and aligned with (u,v).
- To set a view plane we have to specify a *view-plane normal vector,* N, and a *view-up vector,* Up, (both, in world coordinates):

Reminder: Vector Product



$$U \times V = \hat{n}|U||V|\sin\theta$$

$$U \times V = \begin{vmatrix} \hat{x} & \hat{y} & \hat{z} \\ u_x & u_y & u_z \\ v_x & v_y & v_z \end{vmatrix} = \begin{bmatrix} u_y v_z - u_z v_y \\ u_z v_x - u_x v_z \\ u_x v_y - u_y v_x \end{bmatrix}$$



- $P_0=(x_0,y_0,z_0)$ is a point where a camera is located.
- P is a point to look-at.
- $N=(P_0-P)/|P_0-P|$ is the view-plane normal vector.
- Up is the view up vector, whose projection onto the view-plane is directed up.

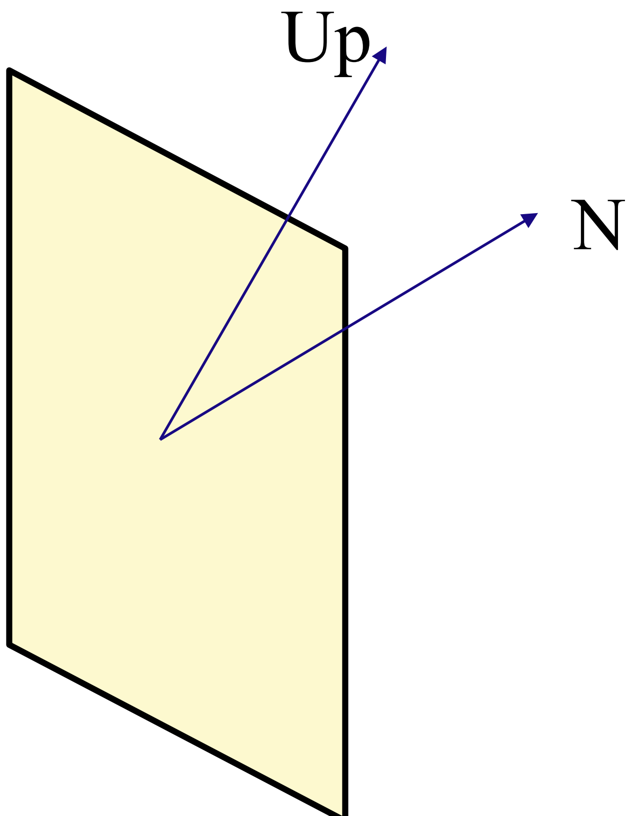
- How to form Viewing coordinate system :

$$w = \frac{N}{|N|} \quad ; \quad u = \frac{Up \times N}{|Up \times N|} \quad ; \quad v = w \times u$$

- The transformation, M, from world-coordinate into viewing-coordinates is:

$$M = \begin{bmatrix} u_x & u_y & u_z & 0 \\ v_x & v_y & v_z & 0 \\ w_x & w_y & w_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & -x_0 \\ 0 & 1 & 0 & -y_0 \\ 0 & 0 & 1 & -z_0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

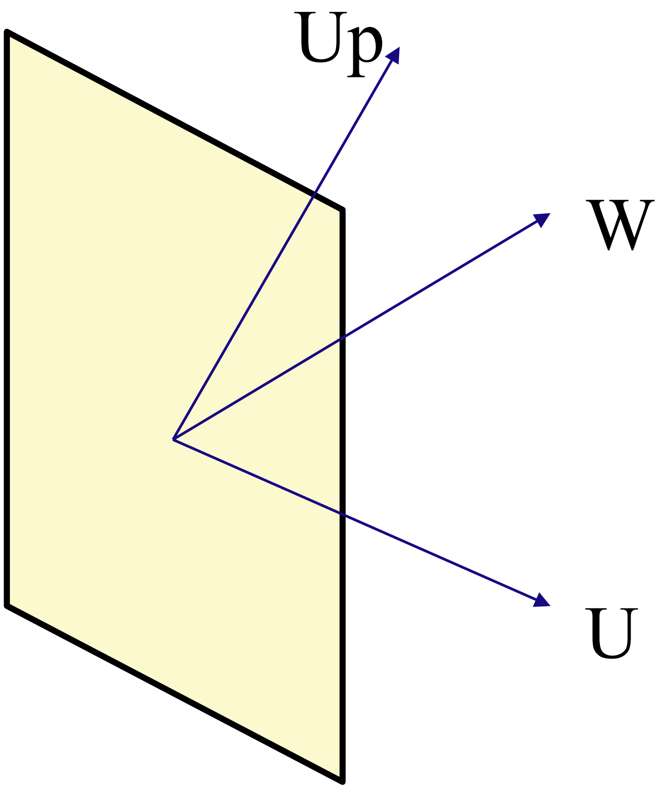
How to form Viewing coordinate system



$$w = N / \| N \|$$

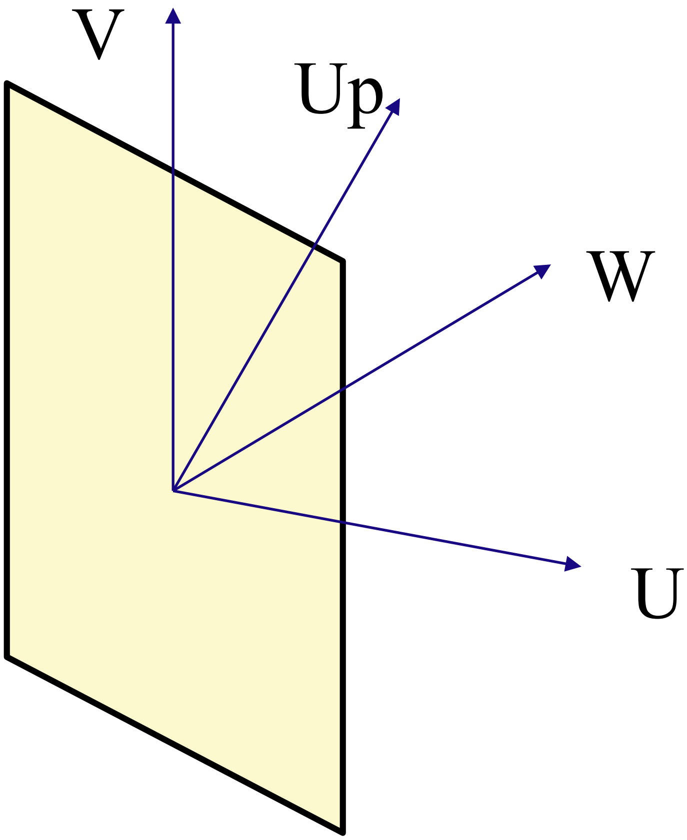
First, normalize the look-at vector to form the w-axis

Create U perpendicular to Up and W



$$U = \frac{Up \times W}{|Up \times W|}$$

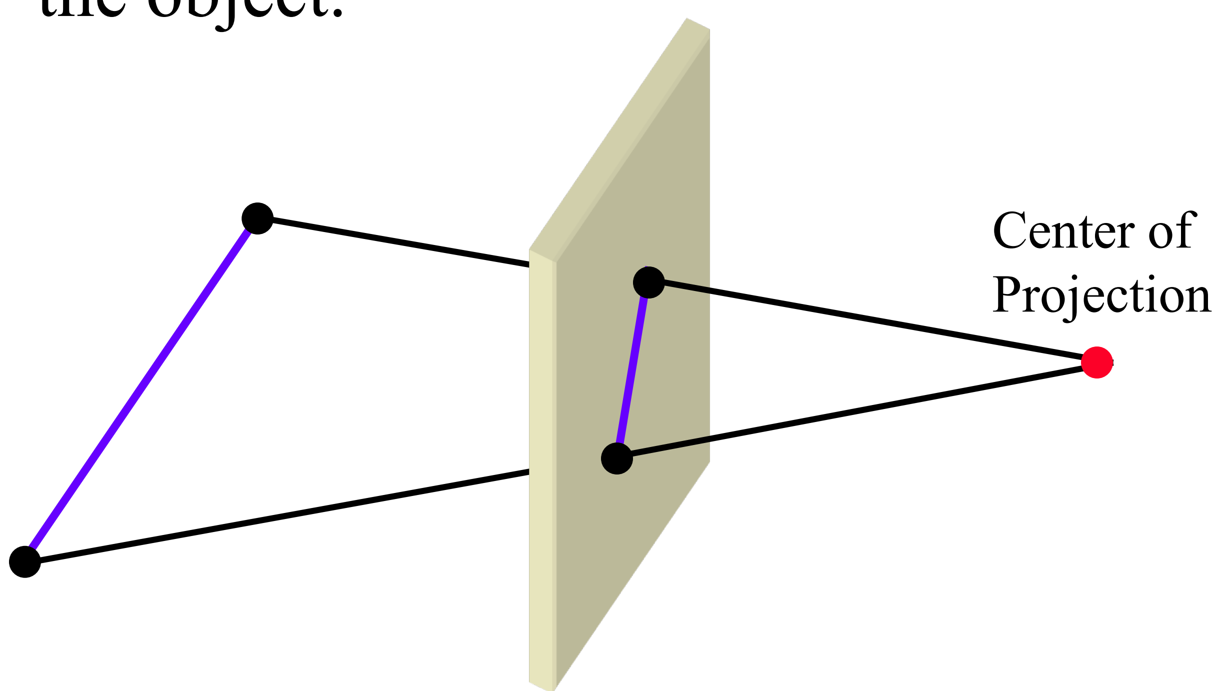
Create V perpendicular to U and W



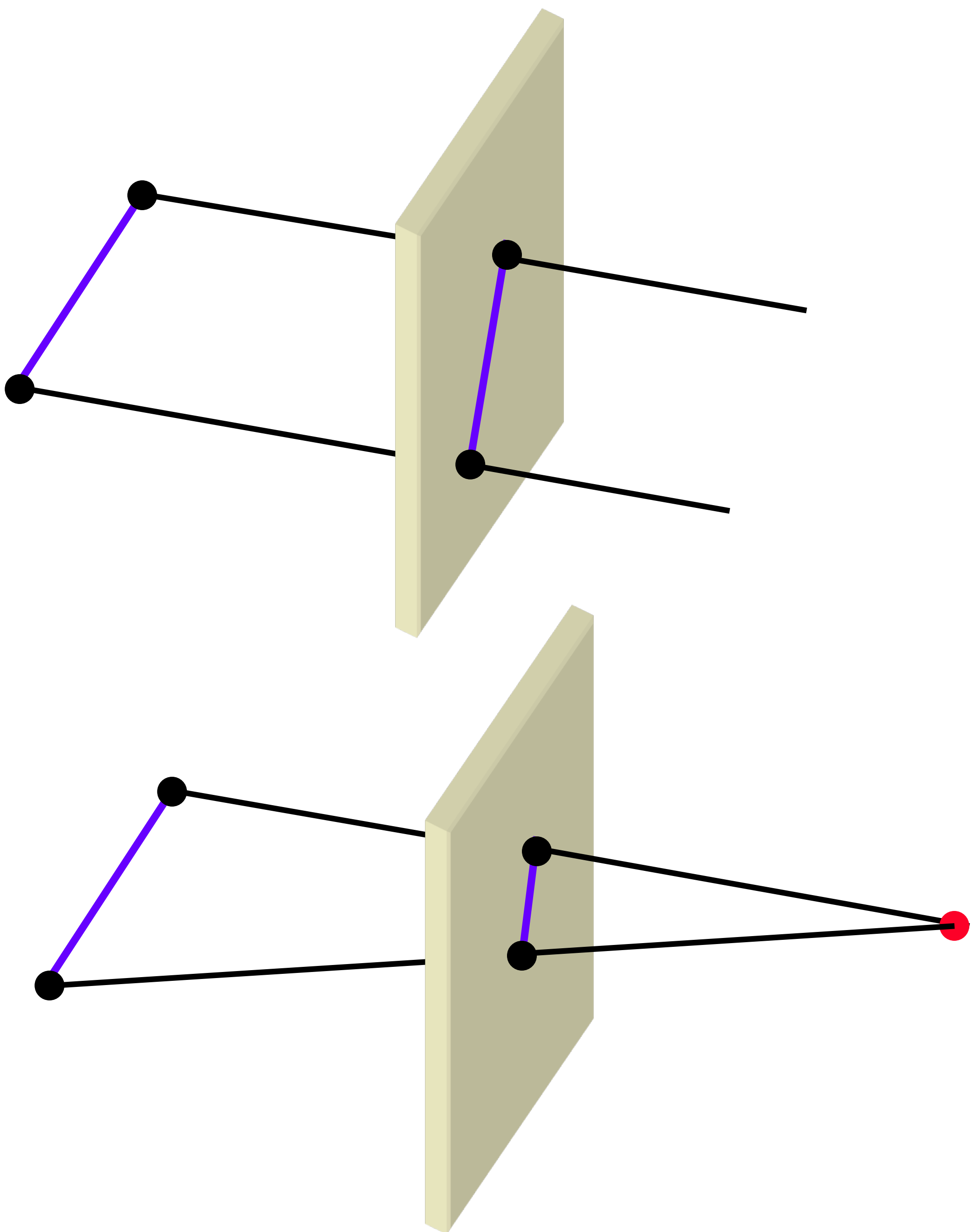
$$V = W \times U$$

Projections

- Viewing 3D objects on a 2D display requires a mapping from 3D to 2D.
- A projection is formed by the intersection of certain lines (*projectors*) with the view plane.
- Projectors are lines from the *center of projection* through each point in the object.

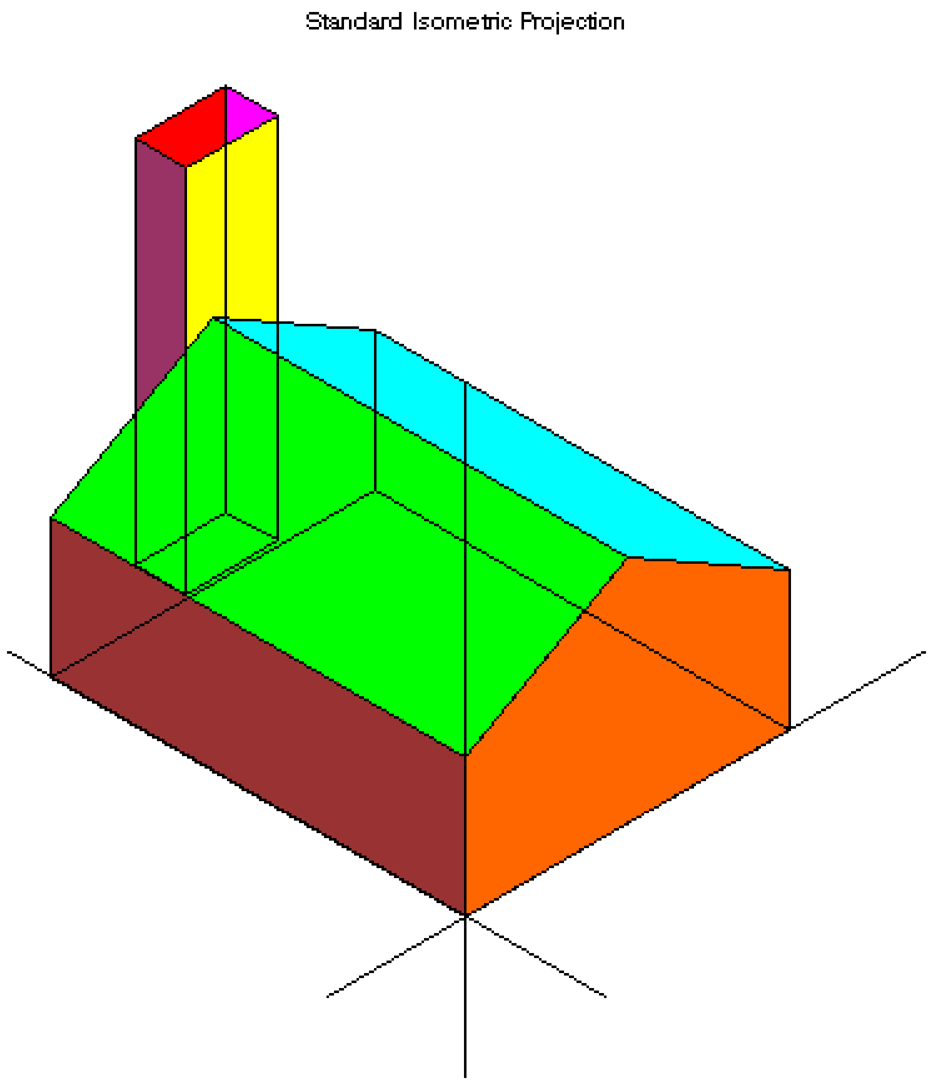


- Center of projection at infinity results with a parallel projection.
- A finite center of projection results with a perspective projection.



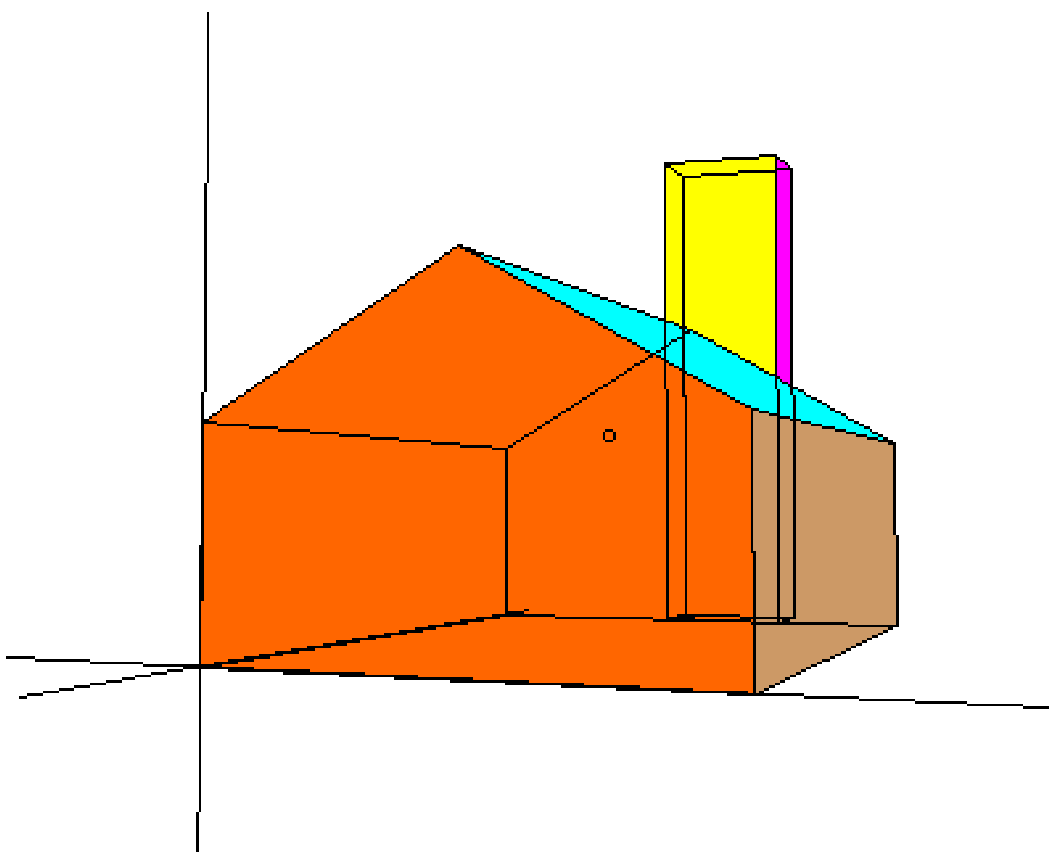
Parallel Projection

A parallel projection preserves relative proportions of objects, but does not give realistic appearance (commonly used in engineering drawing).



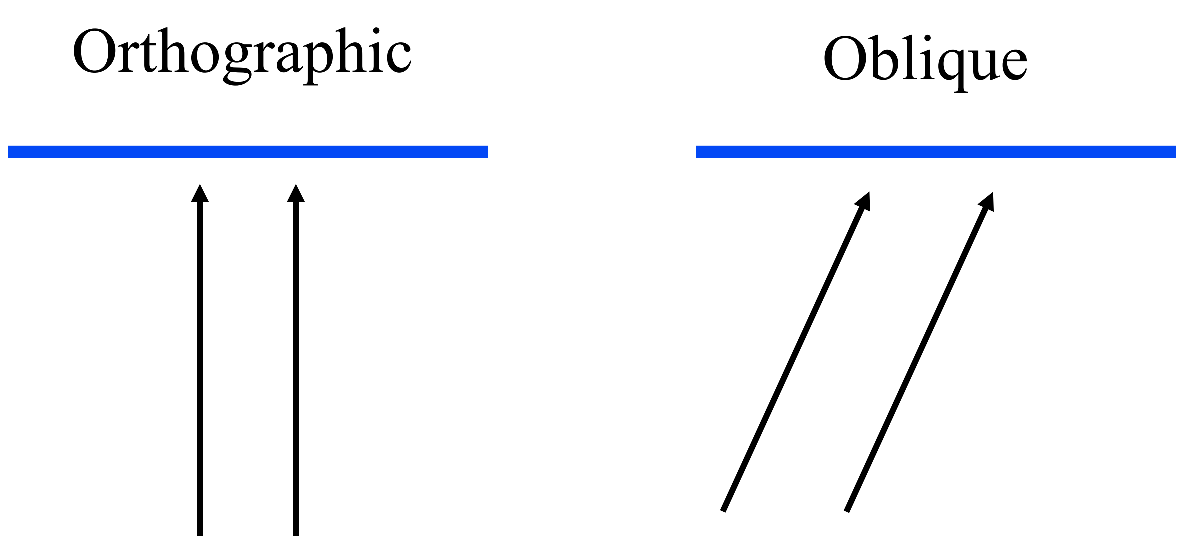
Perspective Projection

A perspective projection produces realistic appearance, but does not preserve relative proportions.



Parallel Projection

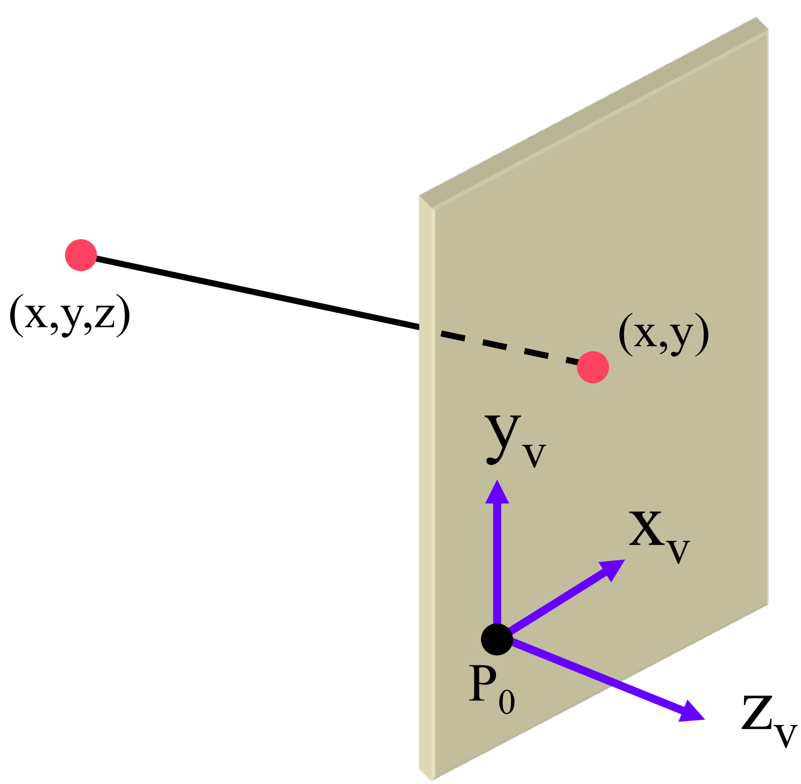
- Projectors are all parallel.
- Orthographic: Projectors are perpendicular to the projection plane.
- Oblique: Projectors are not necessarily perpendicular to the projection plane.



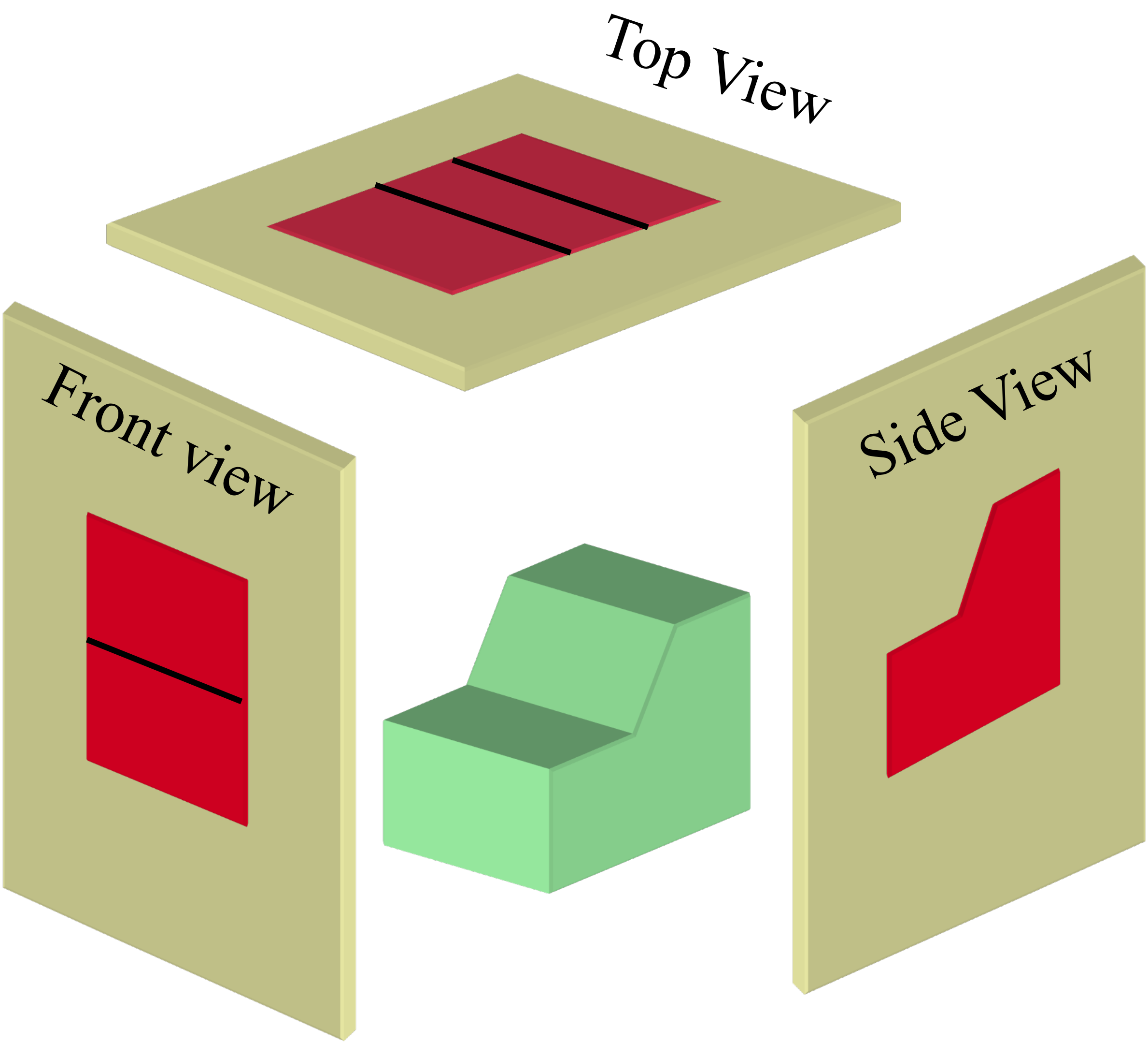
Orthographic Projection

- Since the viewing plane is aligned with (x_v,y_v), orthographic projection is performed by:

$$\begin{bmatrix} x_p \\ y_p \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} x_v \\ y_v \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_v \\ y_v \\ z_v \\ 1 \end{bmatrix}$$

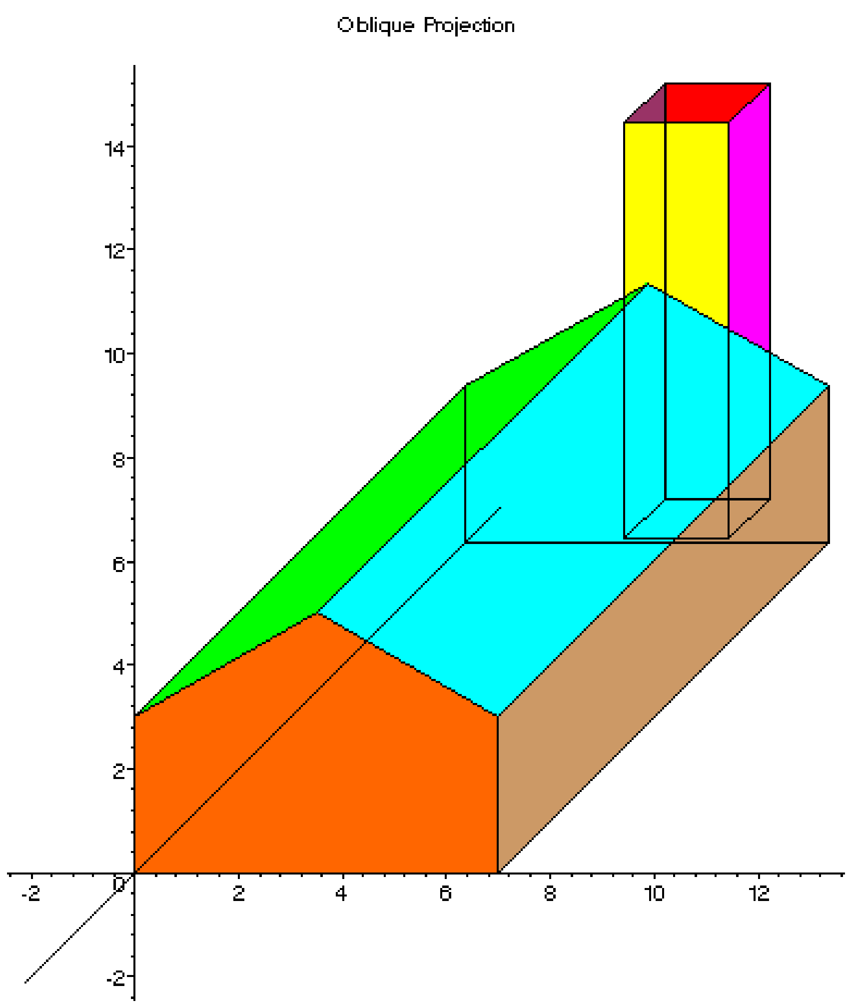


- Lengths and angles of faces parallel to the viewing planes are preserved.
- **Problem**: 3D nature of projected objects is difficult to deduce.

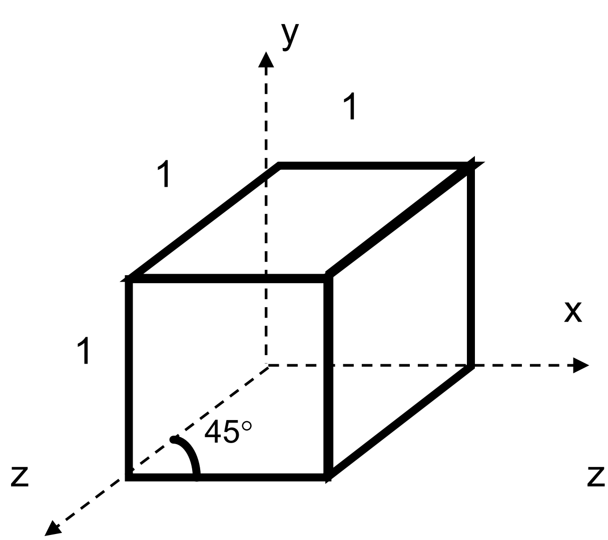


Oblique Projection

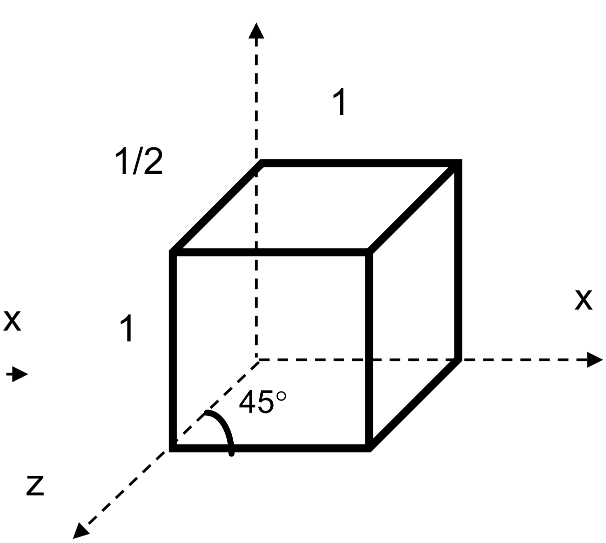
- Projectors are ***not*** perpendicular to the viewing plane.
- Angles and lengths are preserved for faces parallel to the plane of projection.
- Somewhat preserves 3D nature of an object.



- Cavalinear projection :
  - Preserves lengths of lines perpendicular to the viewing plane.
  - 3D nature can be captured but shape seems distorted.
- Cabinet projection:
  - lines perpendicular to the viewing plane project at 1/2 of their length.
  - A more realistic view than the Cavalinear projection.



Cavalinear Projection

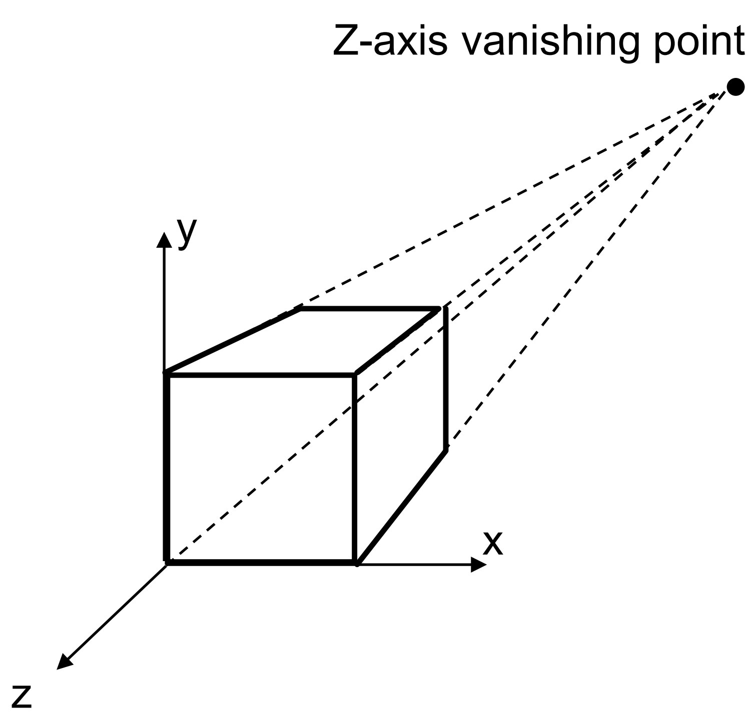


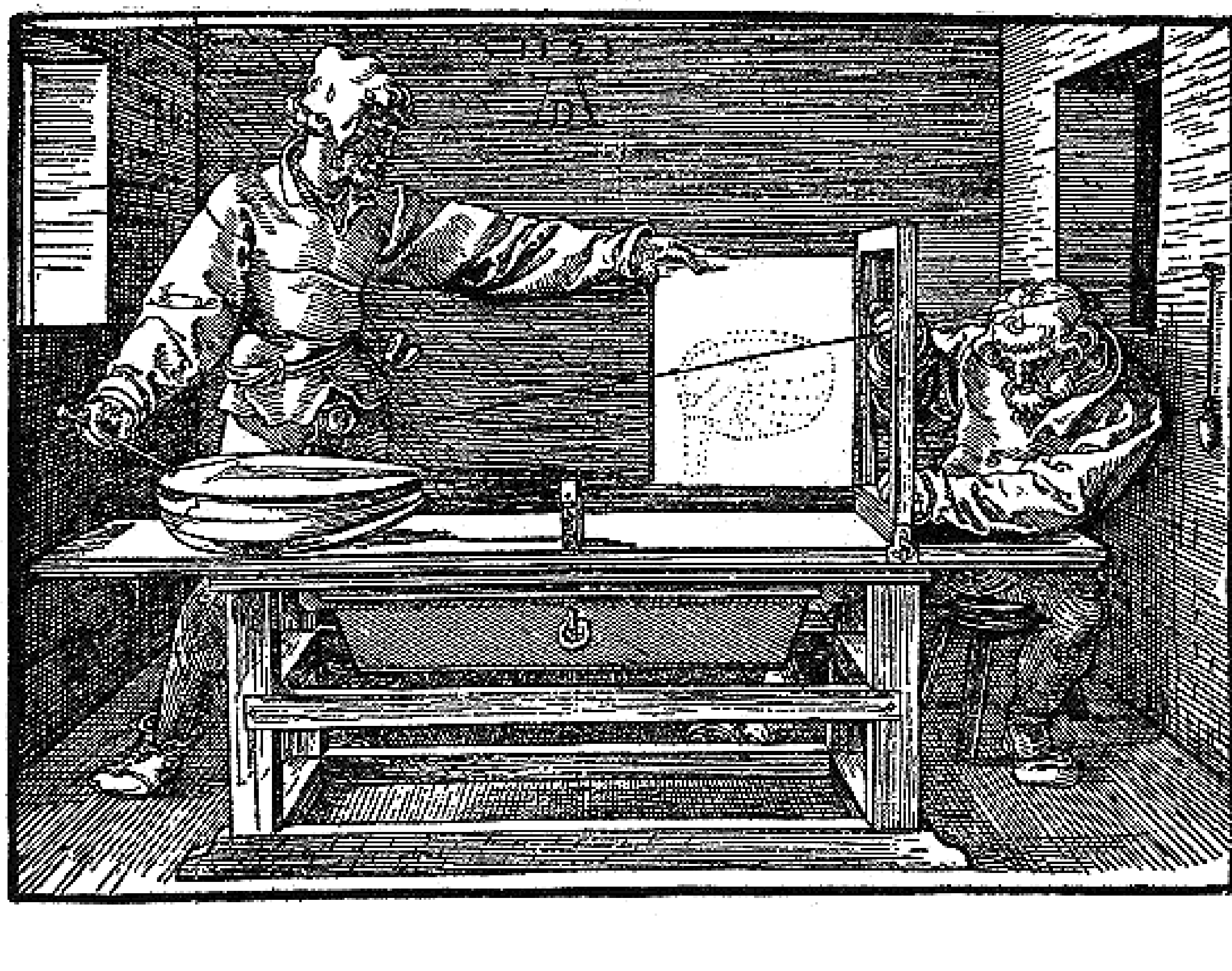
Cabinet Projection

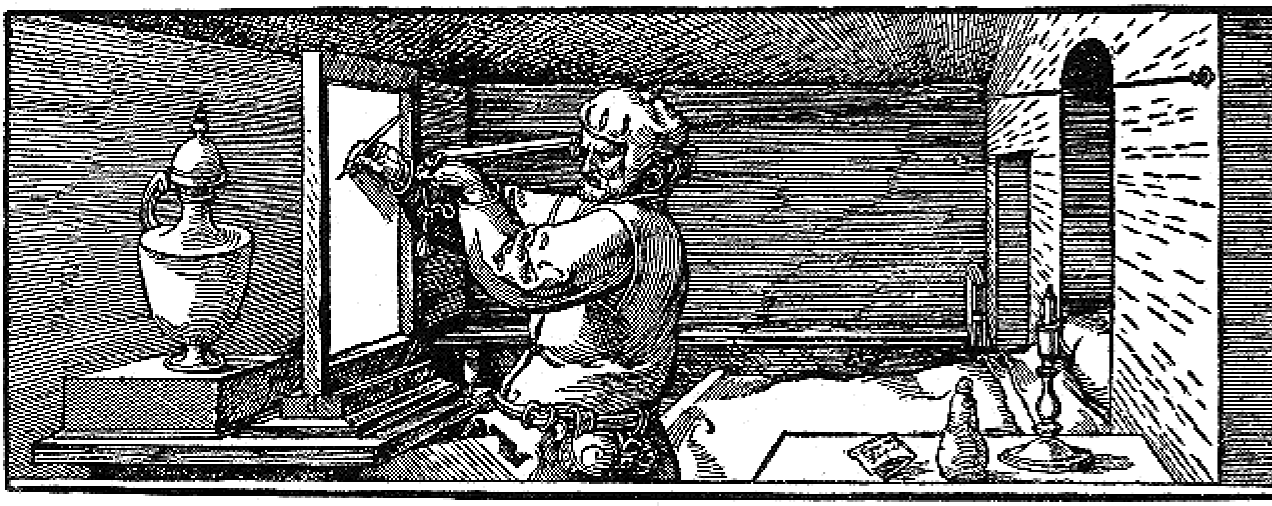
Perspective Projection

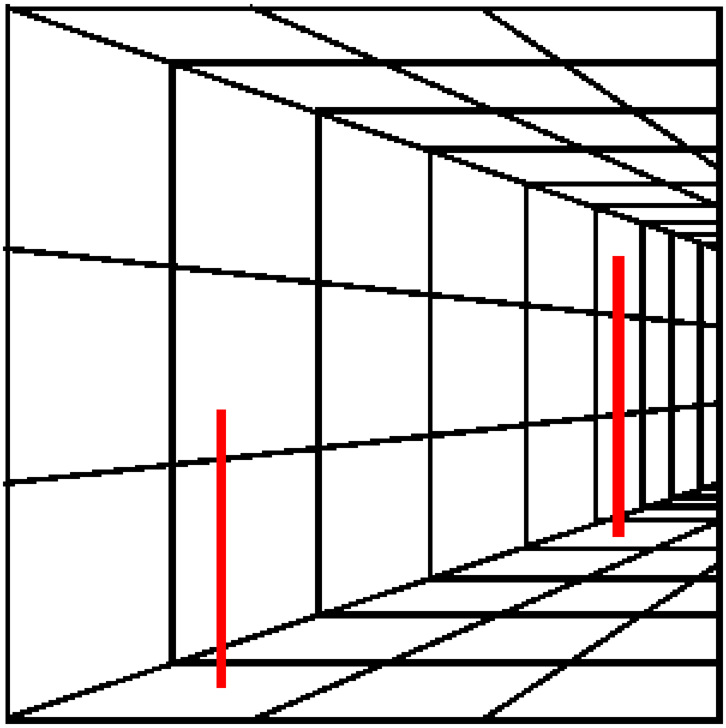
- In a perspective projection, the center of projection is at a finite distance from the viewing plane.
- Parallel lines that are not parallel to the viewing plane, converge to a *vanishing point*.

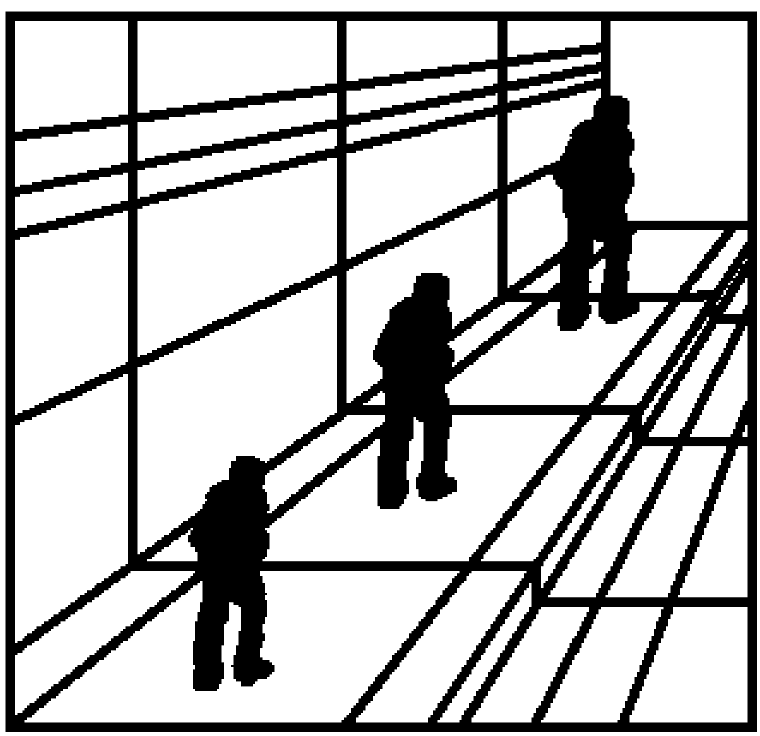
➡ A vanishing point is the projection of a point at infinity.





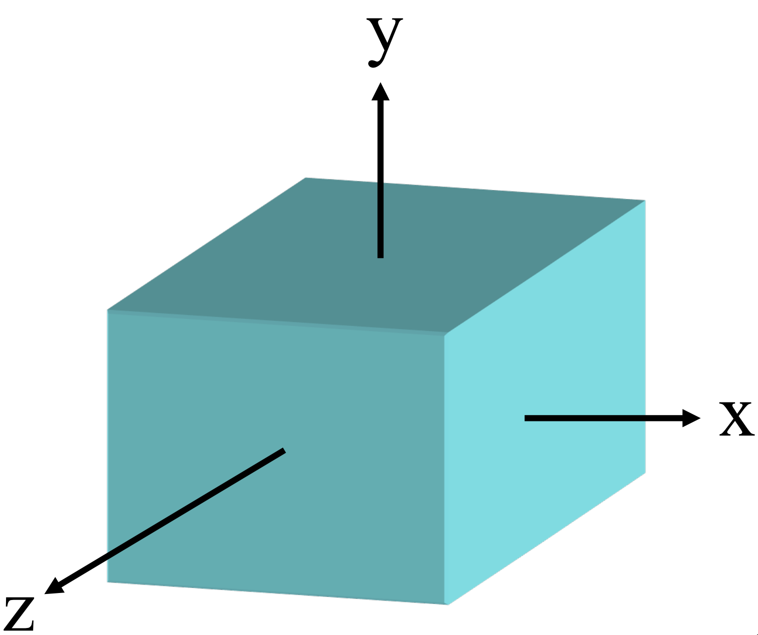


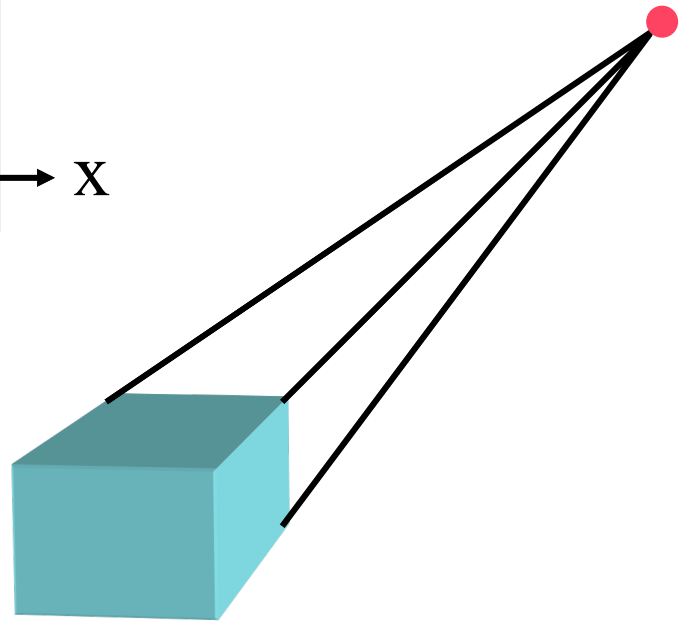




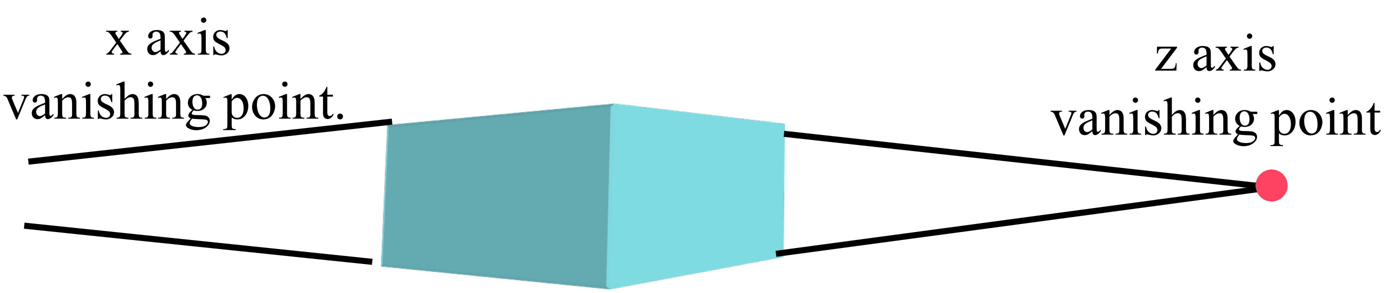
Vanishing Points

- There are infinitely many general vanishing points.
- There can be up to three *axis vanishing points* (principal vanishing points).
- Perspective projections are categorized by the number of principal vanishing points, equal to the number of principal axes intersected by the viewing plane.
- Most commonly used: one-point and two-points perspective.

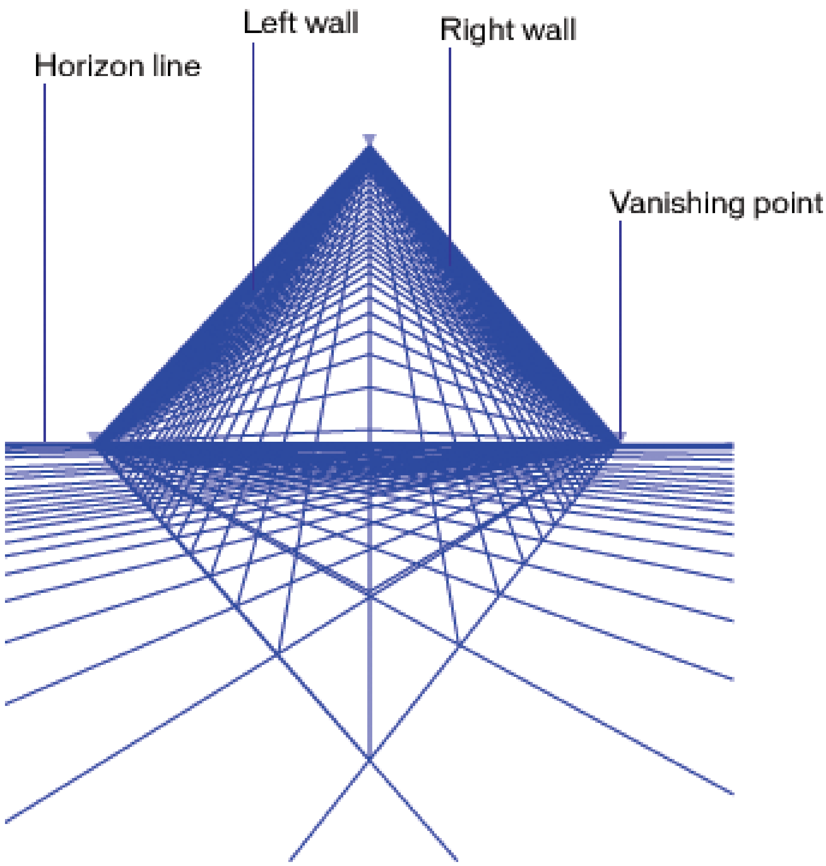




One point (z axis)
perspective projection

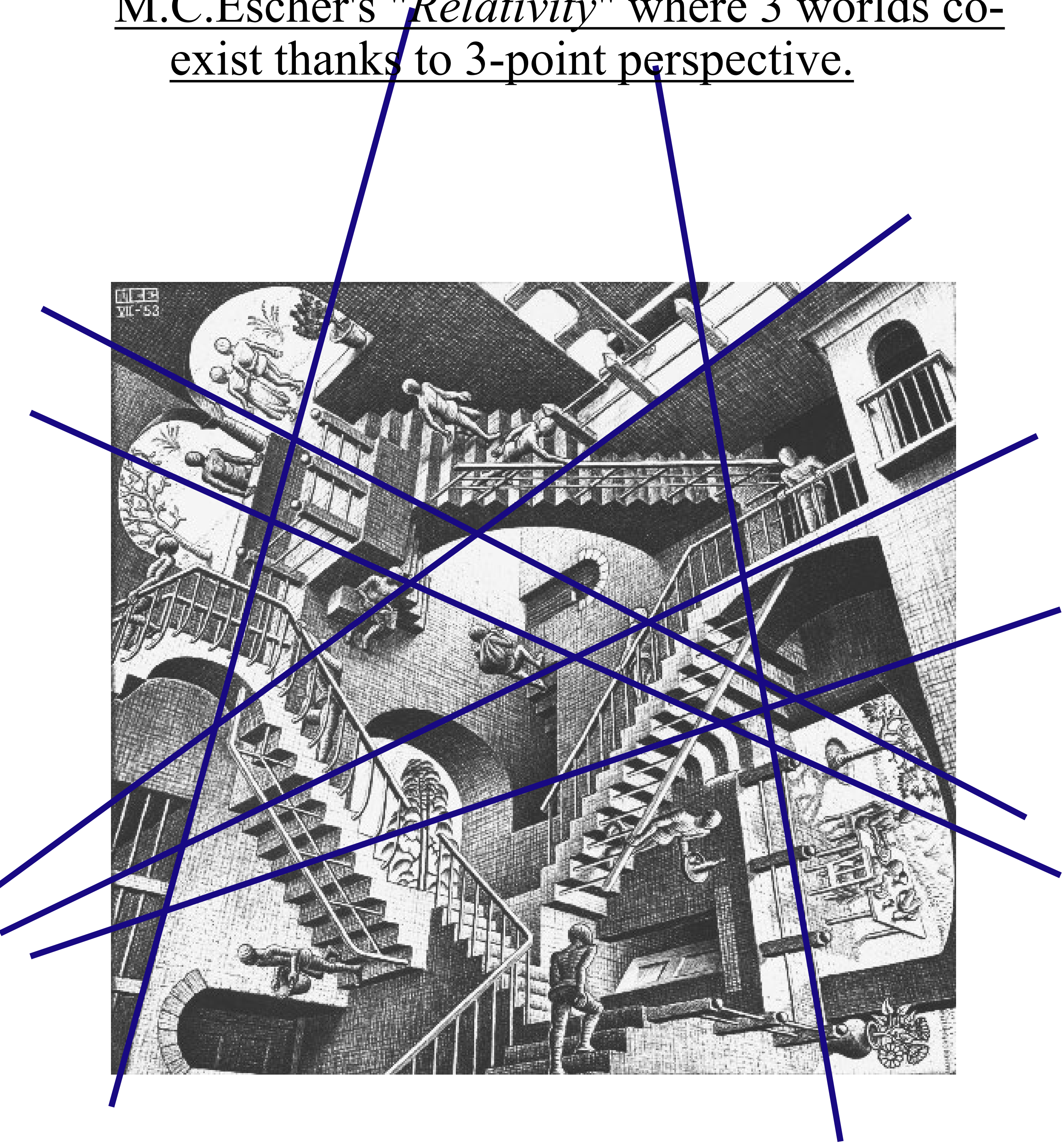


Two points
perspective projection

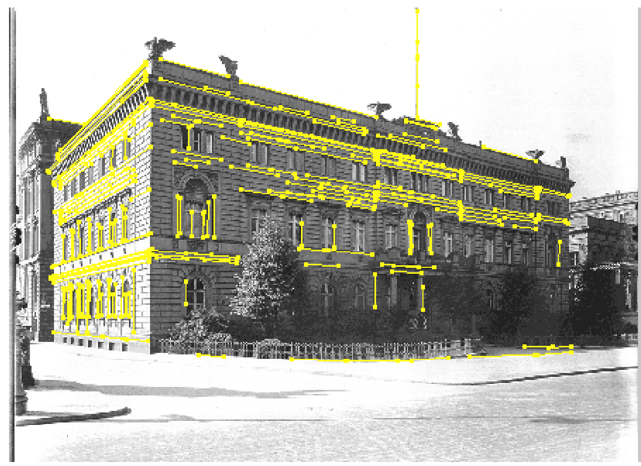
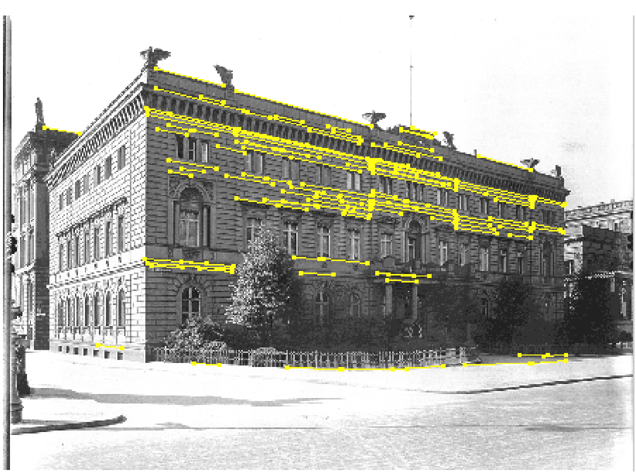

Left wall
Right wall
Horizon line
Vanishing point

3-point Perspective

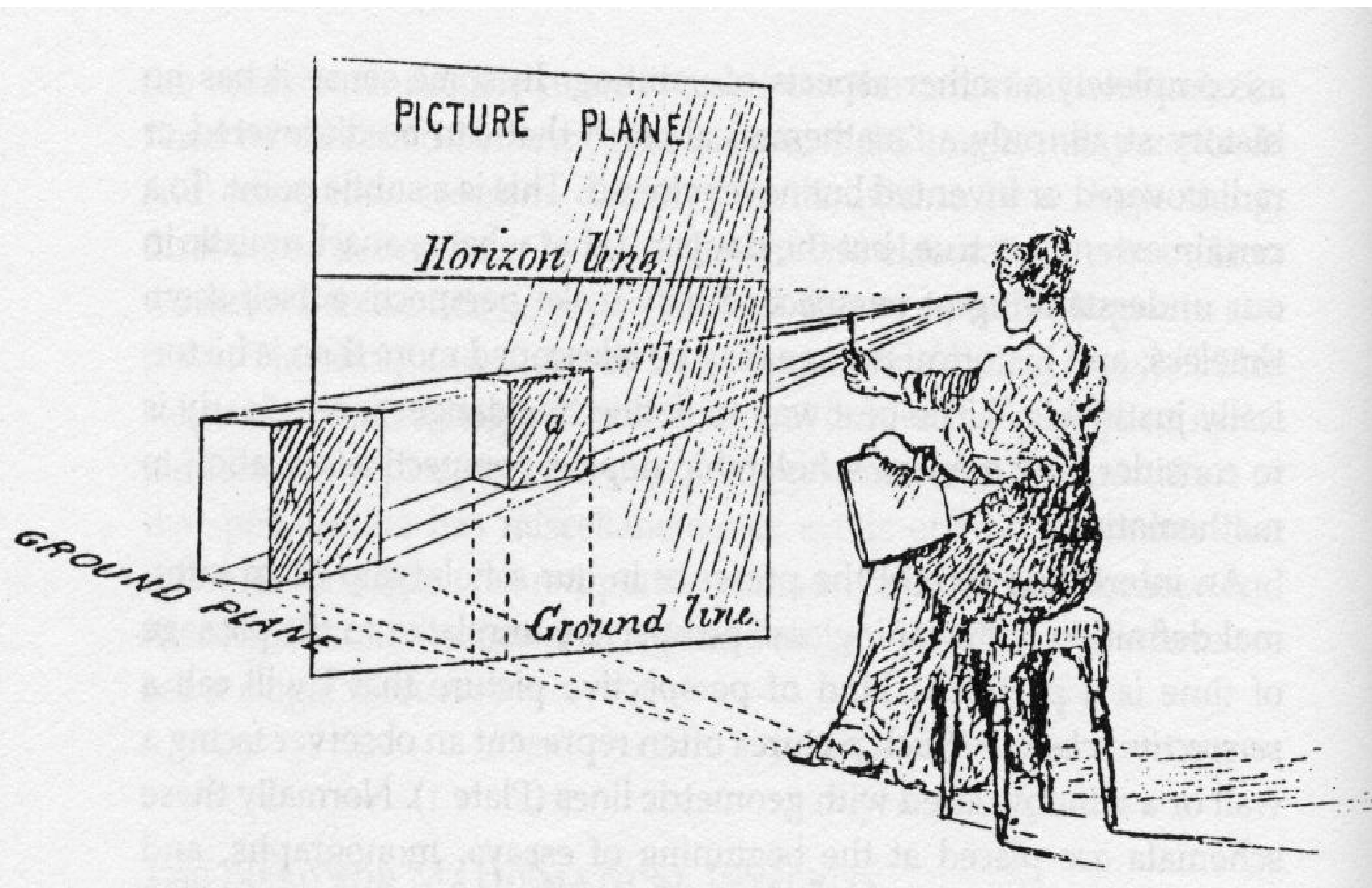
M.C.Escher's "*Relativity*" where 3 worlds co-exist thanks to 3-point perspective.



3-point Perspective

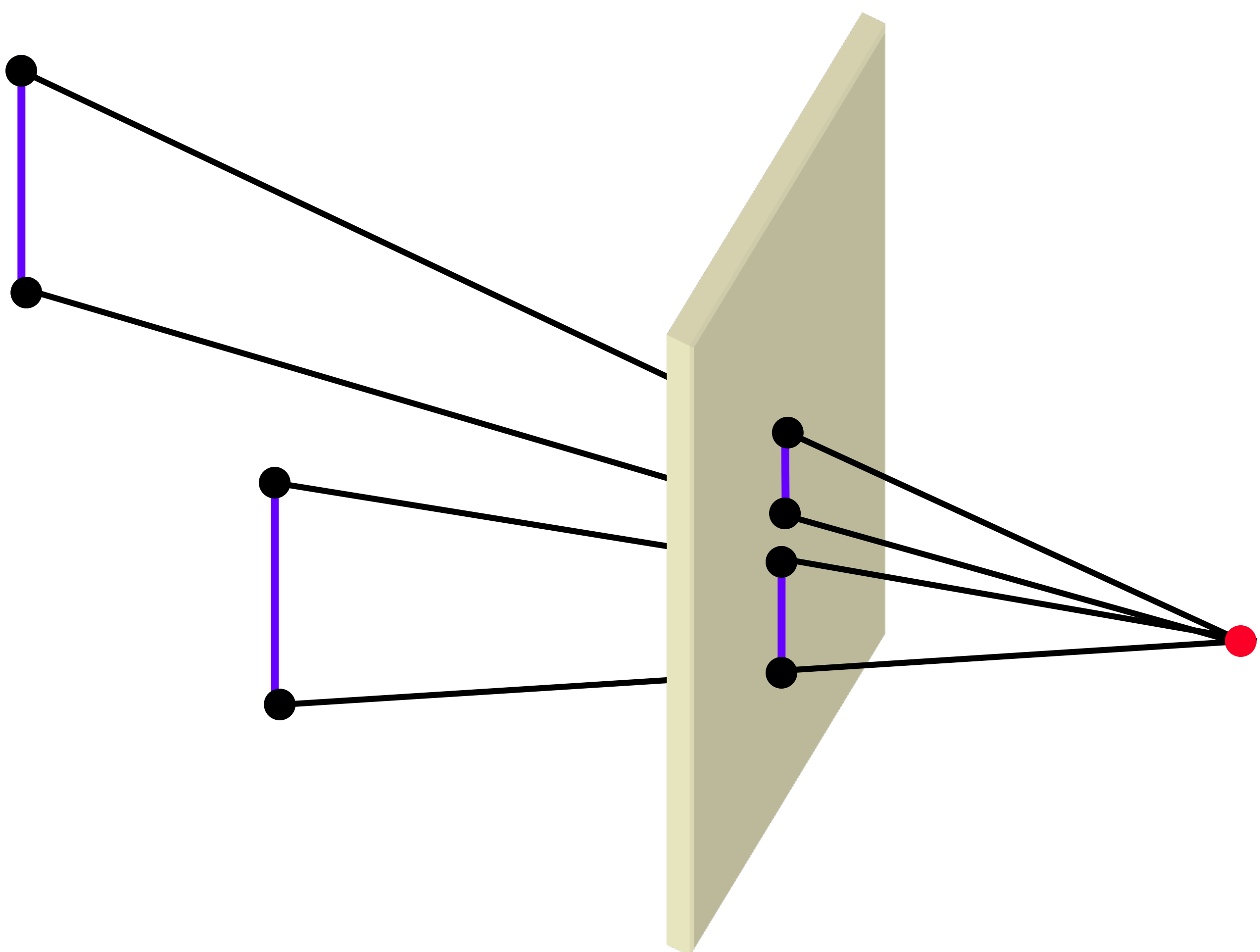




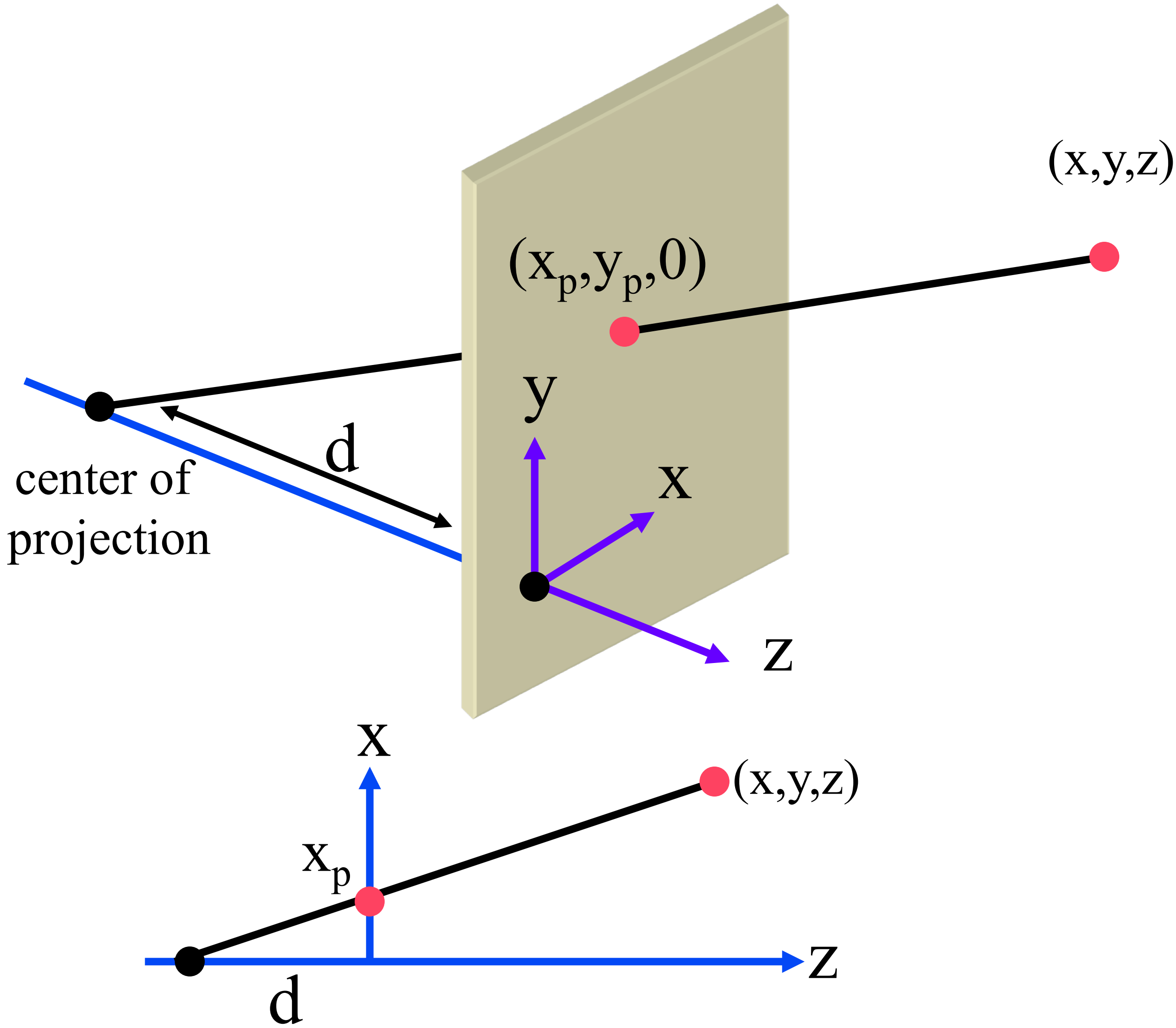

Perspective Projection



https://vimeo.com/48425421

A perspective projection produces realistic appearance, but does not preserve relative proportions.





- Using similar triangles it follows:

$$\frac{x_p}{d} = \frac{x}{z+d} \quad ; \quad \frac{y_p}{d} = \frac{y}{z+d}$$

$$\Rightarrow \quad x_p = \frac{d \cdot x}{z+d} \quad ; \quad y_p = \frac{d \cdot y}{z+d} \quad ; \quad z_p = 0$$

- Thus, a perspective projection matrix is defined:

$$M_{per} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \dfrac{1}{d} & 1 \end{bmatrix}$$

$$M_{per}P = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \dfrac{1}{d} & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} x \\ y \\ 0 \\ \dfrac{z+d}{d} \end{bmatrix}$$

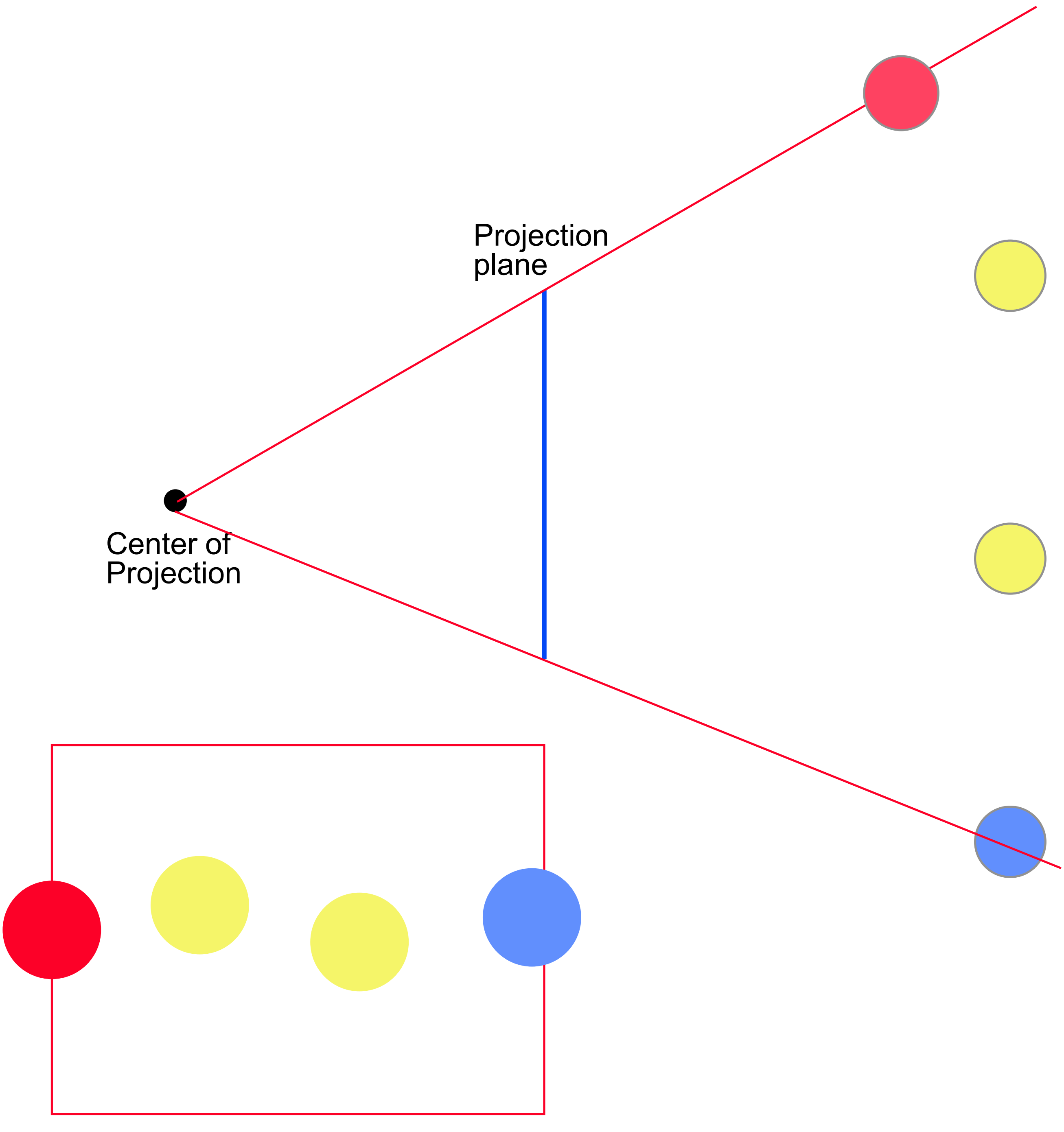
$$\Rightarrow \quad x_p = \frac{d \cdot x}{z+d} \quad ; \quad y_p = \frac{d \cdot y}{z+d} \quad ; \quad z_p = 0$$

Observations

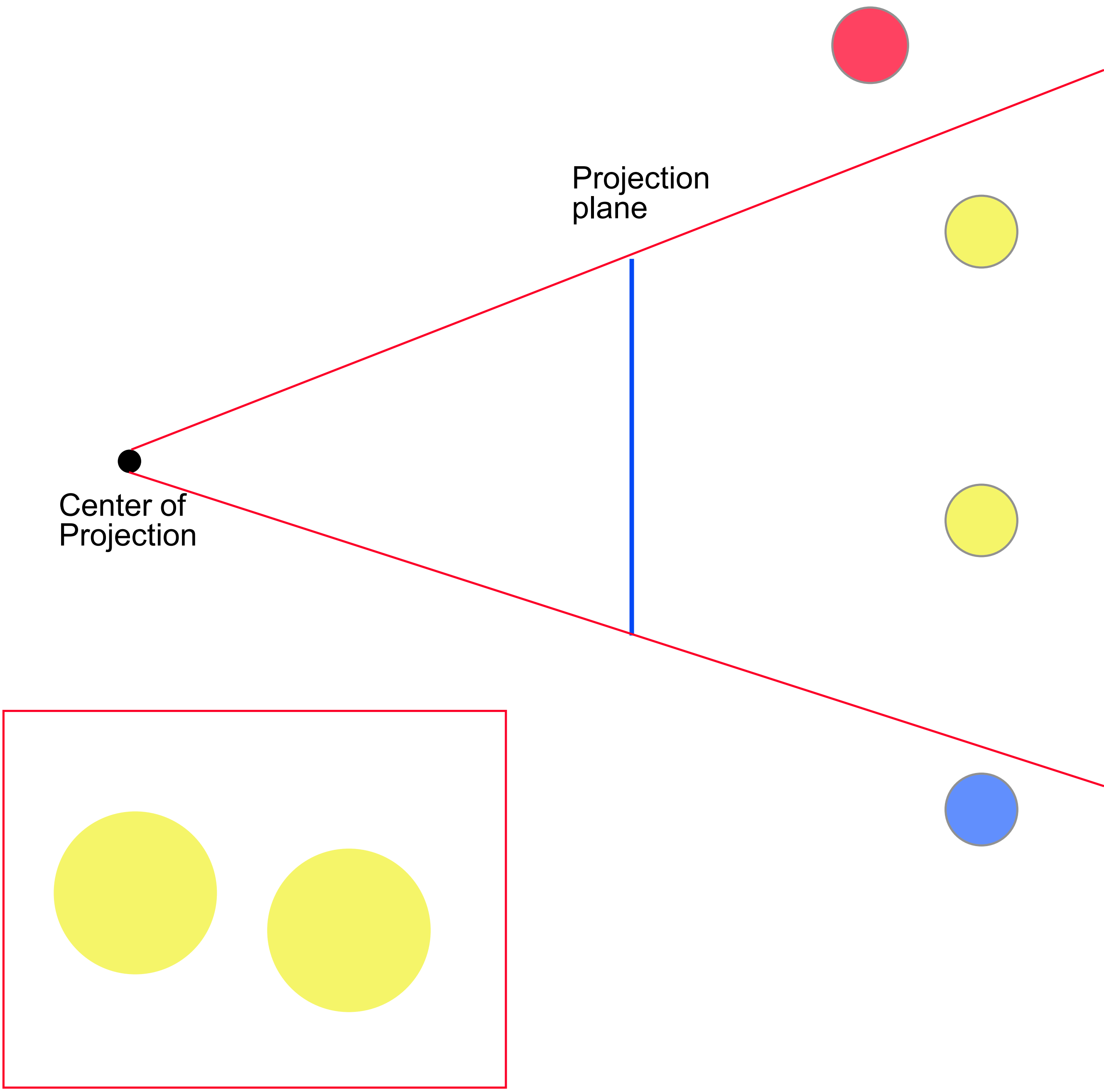
- M_{per} is singular ($|M_{per}|=0$), thus M_{per} is a many to one mapping

- Points on the viewing plane ($z=0$) do not change.
- The vanishing point of parallel lines directed to (U_x,U_y,U_z) is at $[dU_x/U_z, dU_y/U_z]$.
- When $d \to \infty$, $M_{per} \to M_{ort}$

Zoom-in

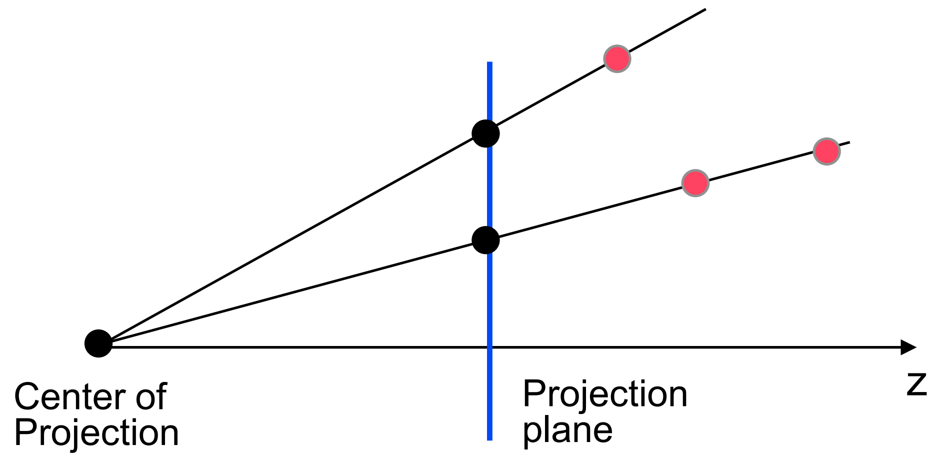


Zoom-in

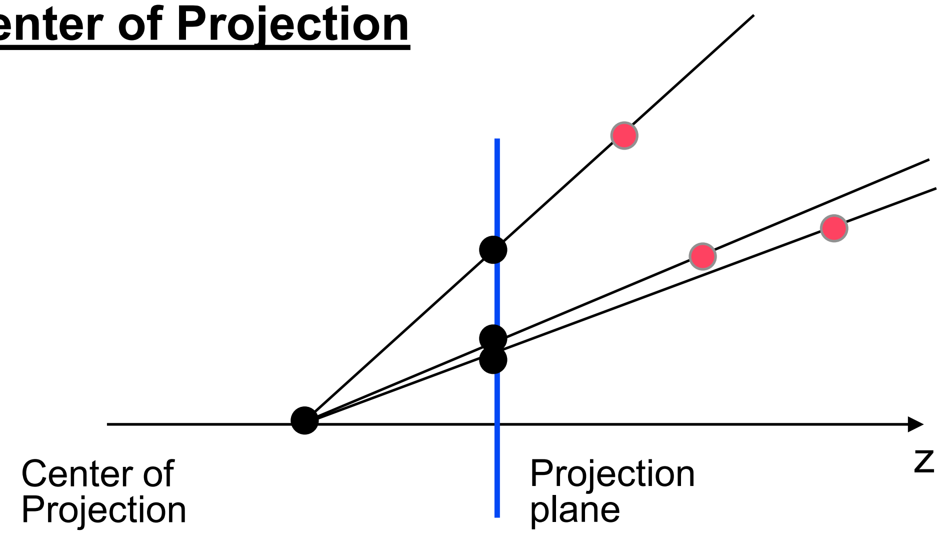


What is the difference between moving the center of projection and moving the projection plane?

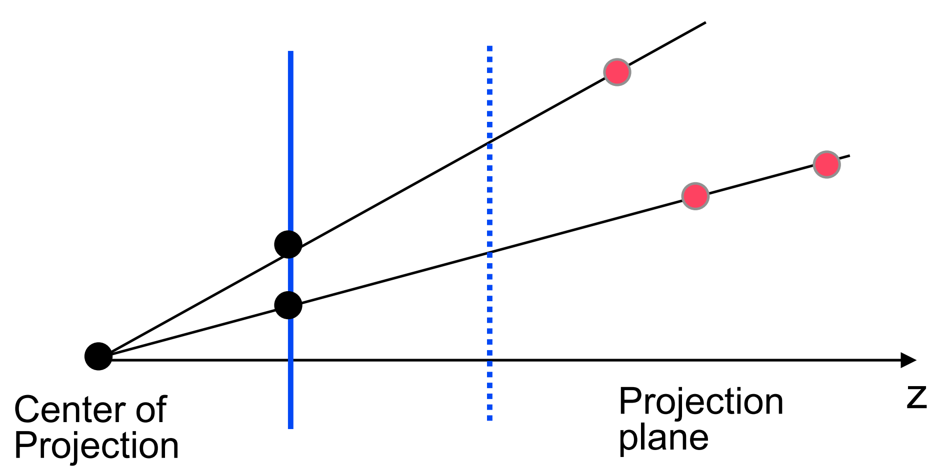
Original



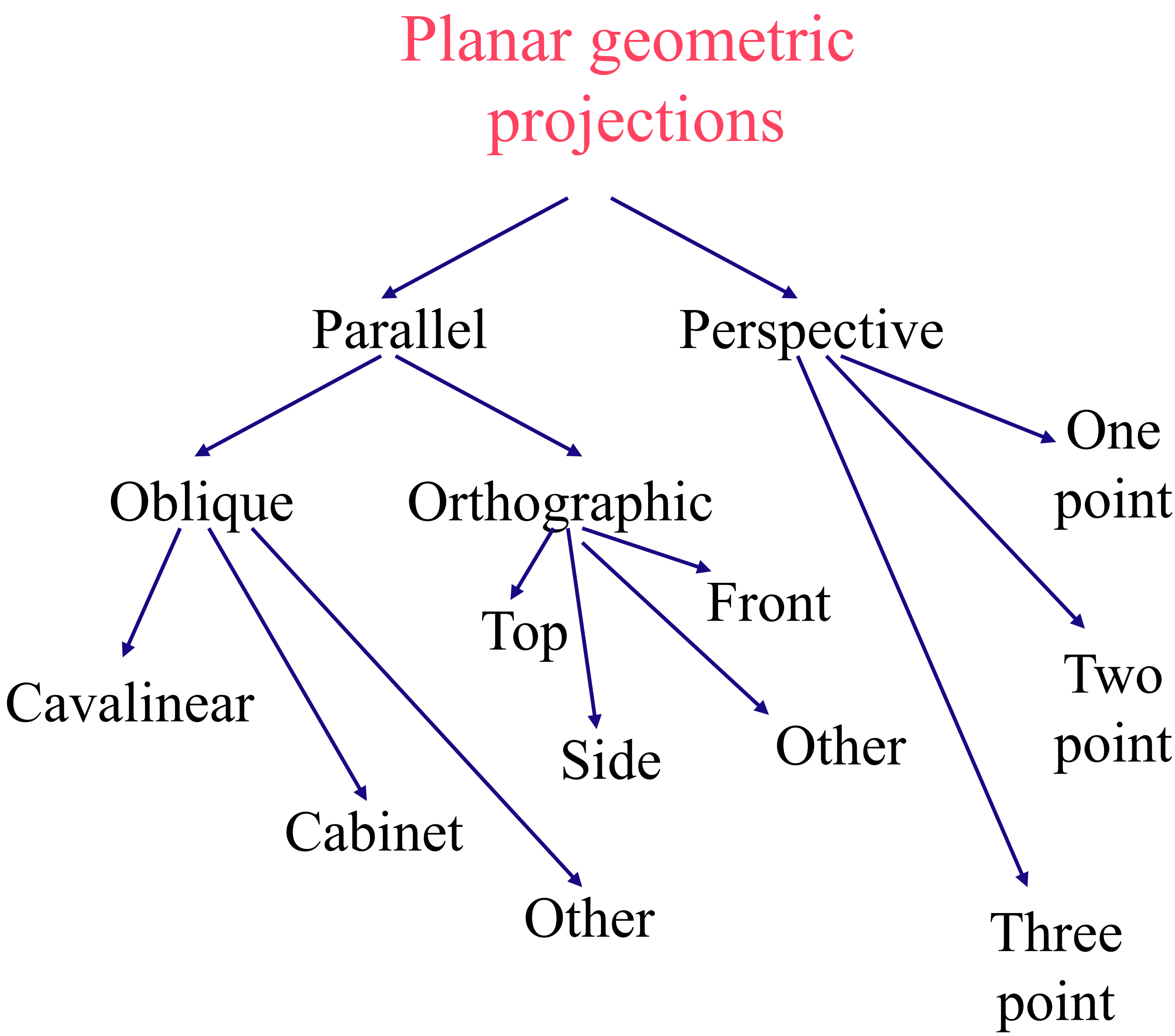
Moving the Center of Projection



Moving the Projection Plane

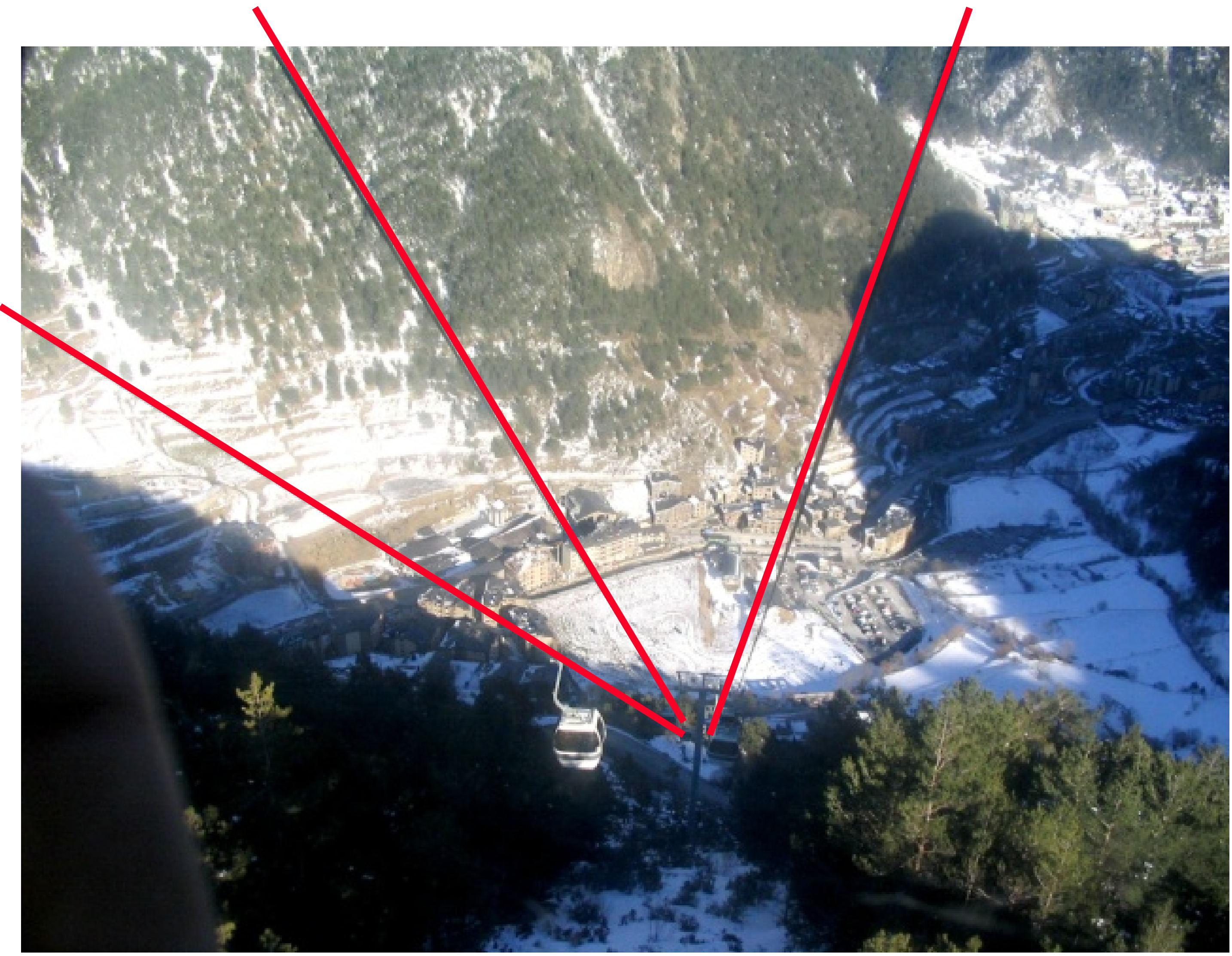


Summary

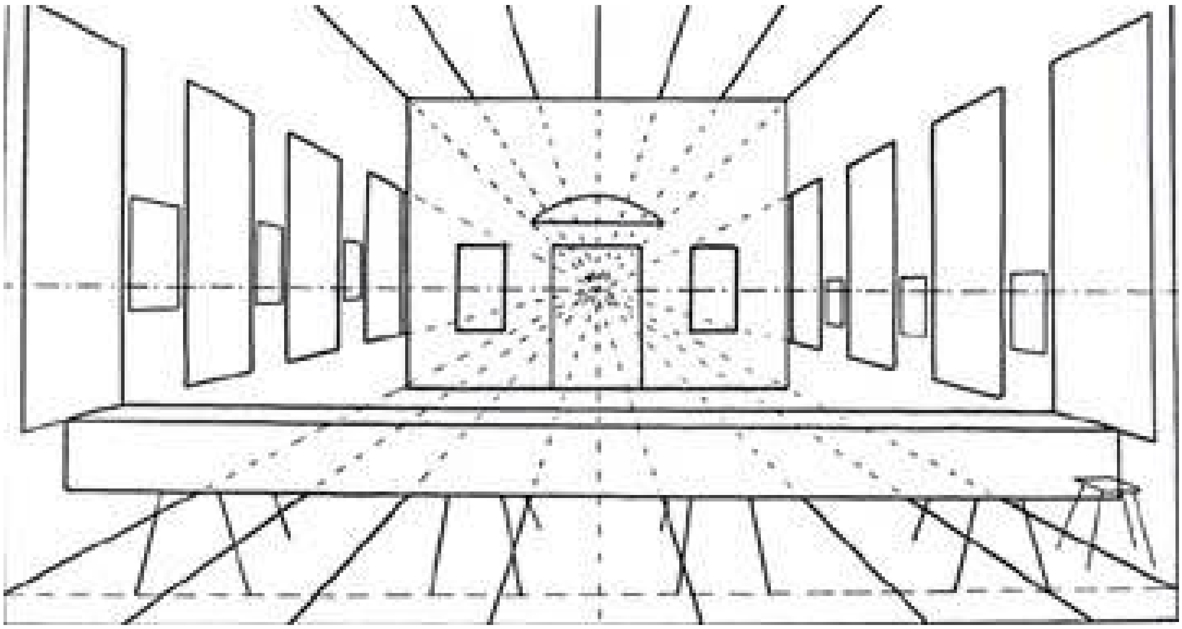


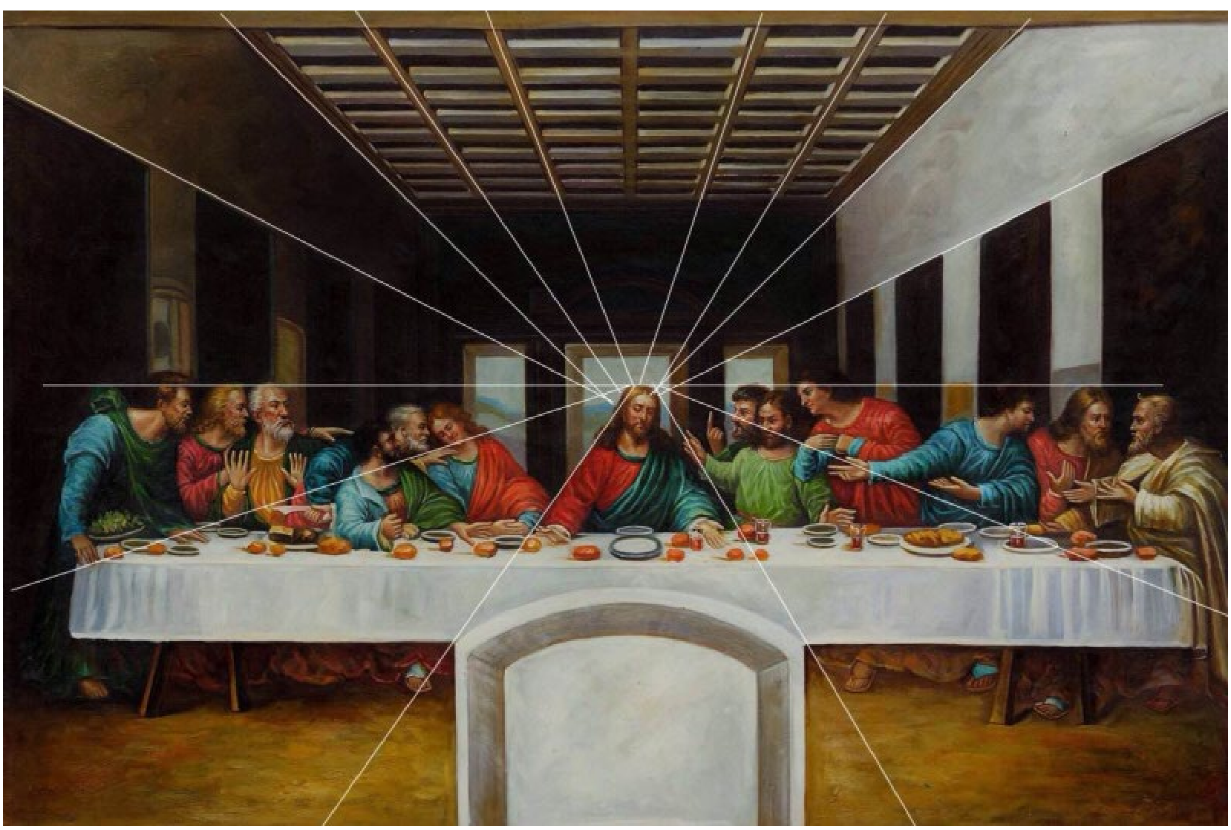
Planar geometric projections
Parallel
Perspective
Oblique
Orthographic
One point
Top
Front
Cavalinear
Two point
Side
Other
Cabinet
Other
Three point

Another view in Perspective



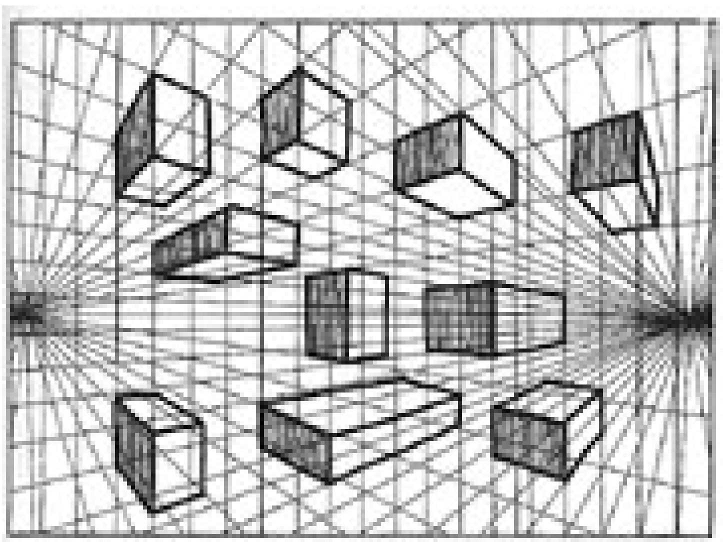
Leonardo da Vinci 1498–1495



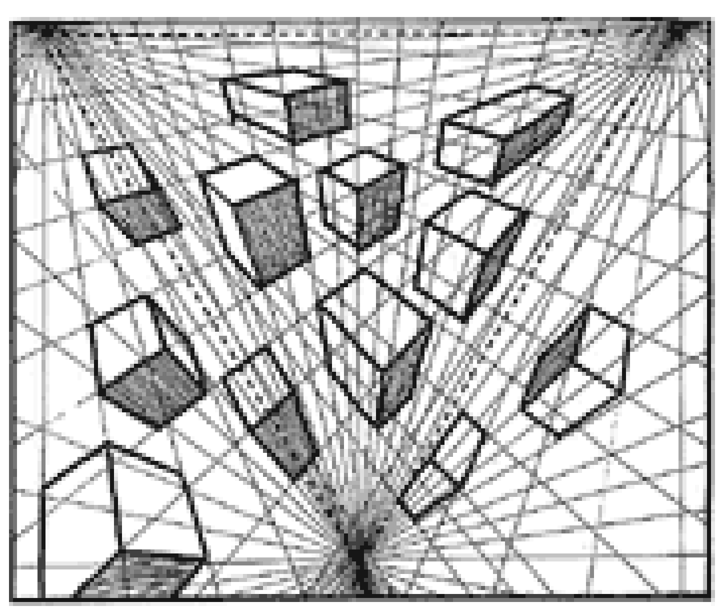


Another view in Perspective

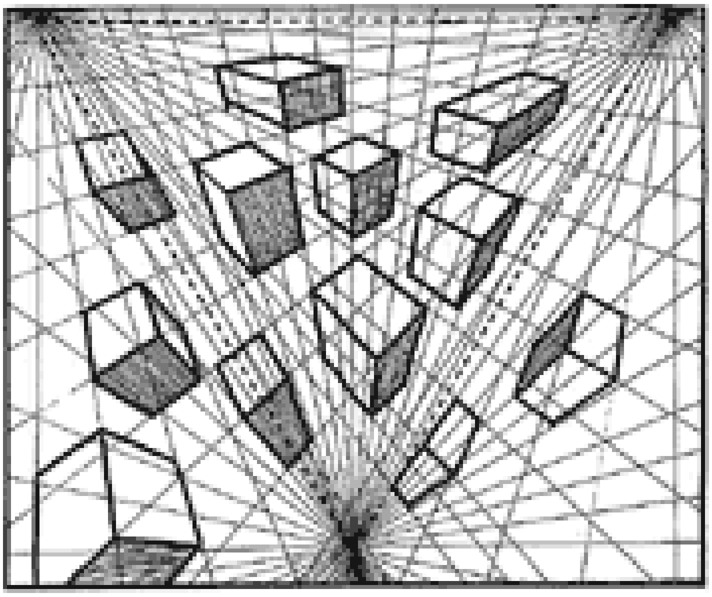
Two-point perspective



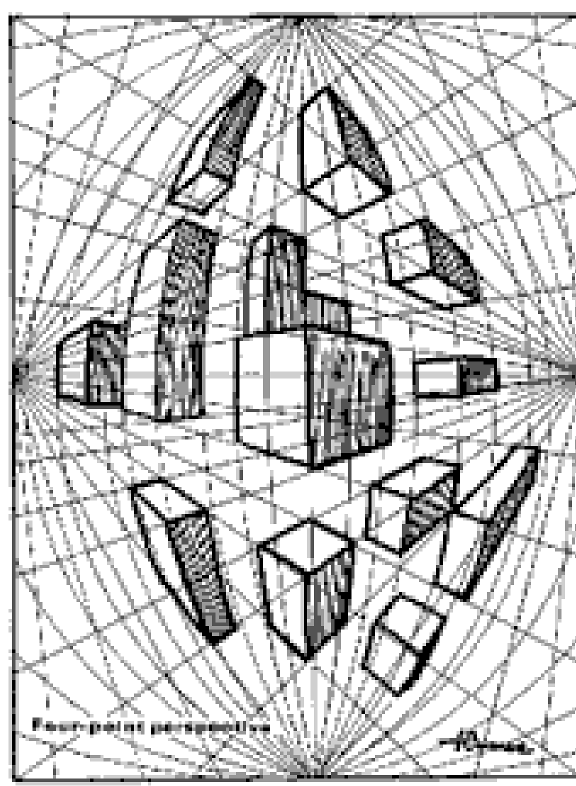
Three-point perspective



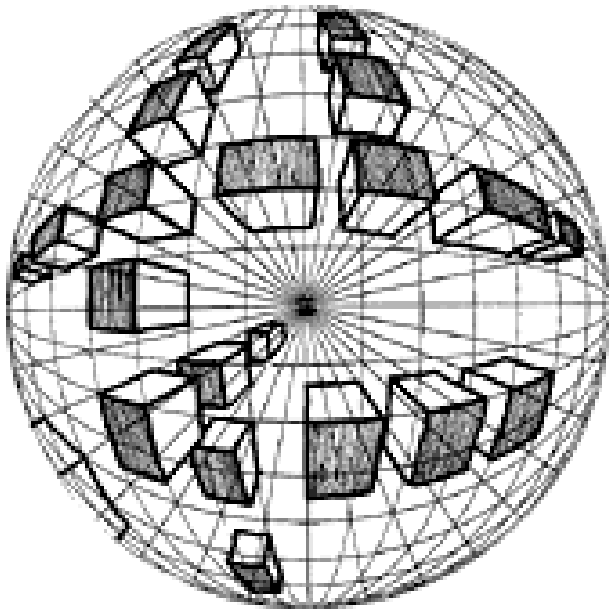
Three-point perspective



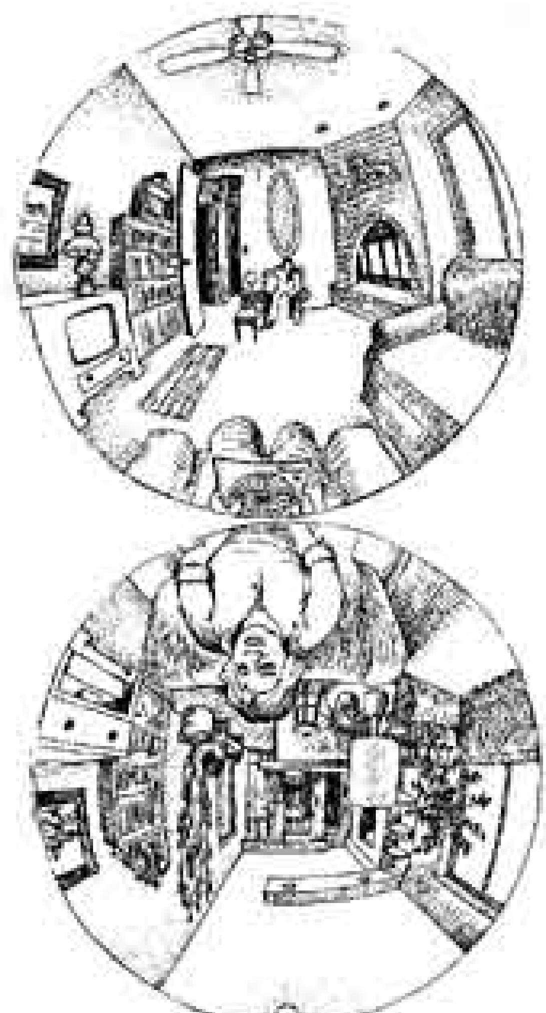
Four-point perspective



Five-point perspective ?



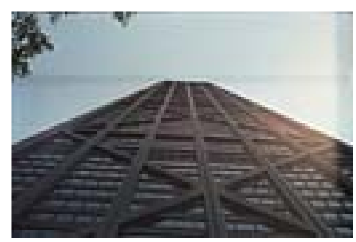
Six-point perspective ?



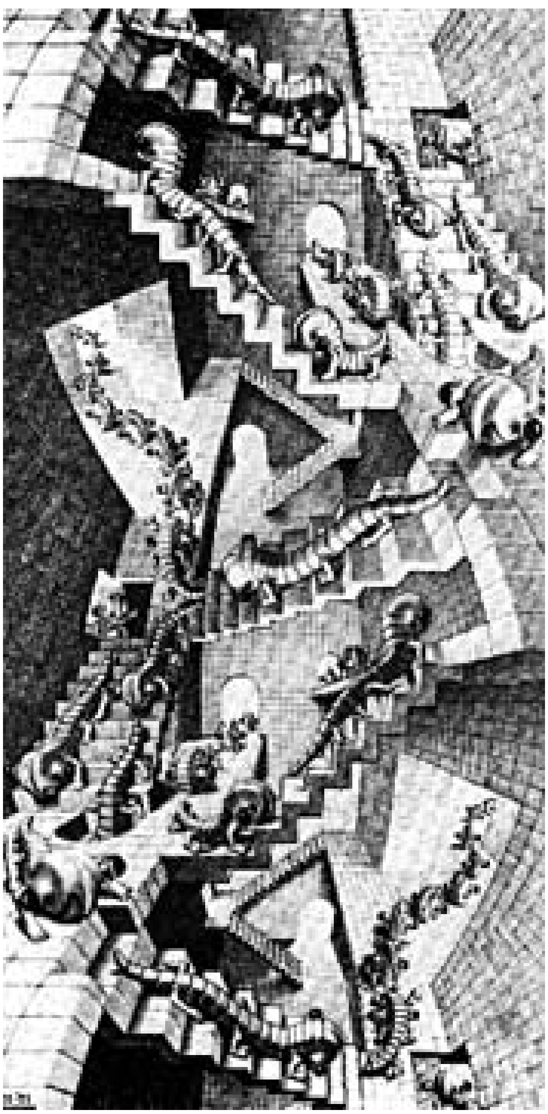
Fisheye views of the Hagia Sophia (Istanbul) (also known as Aya Sofya)

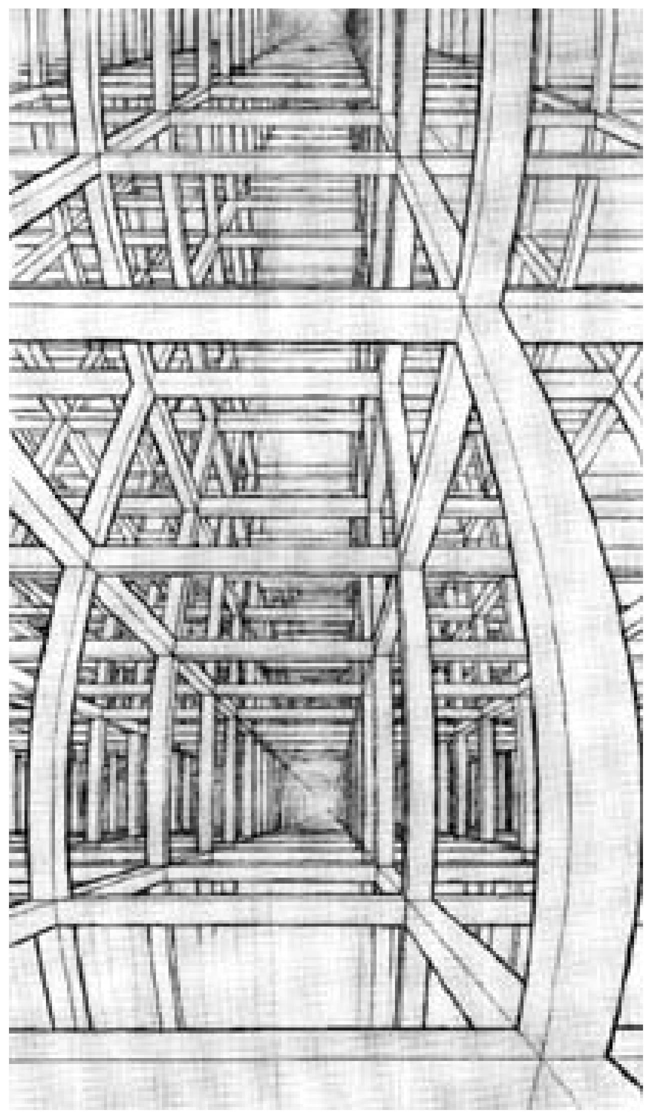
Fisheye view

Vertical lines

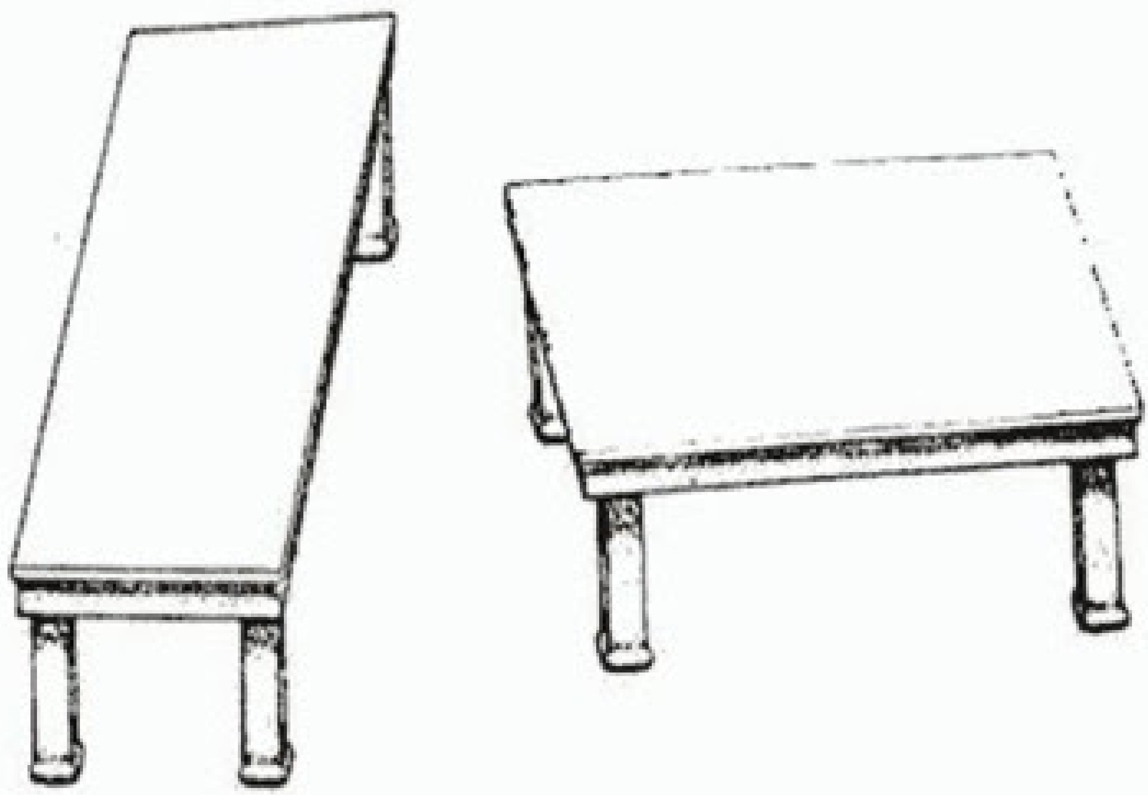


House of Stairs



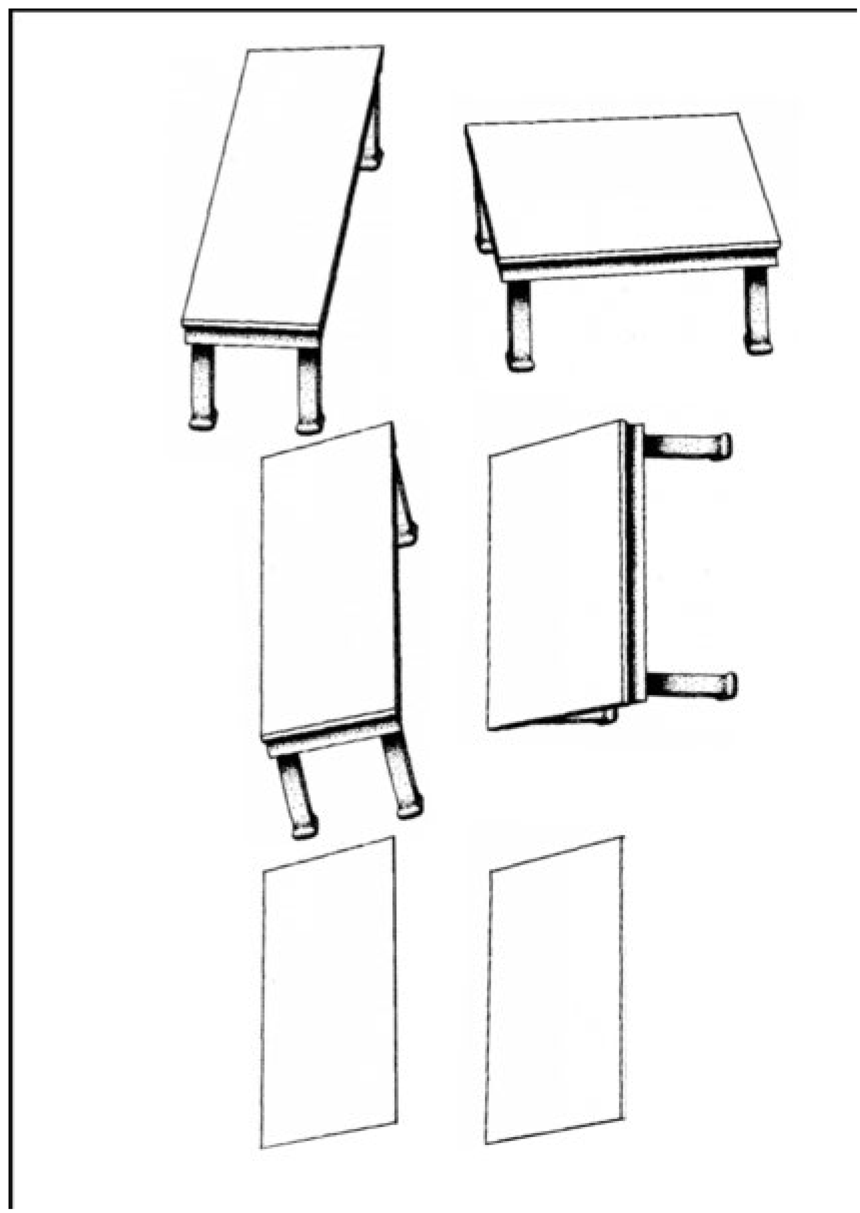


M.C. Escher's House of Stairs



Shepard's Table Illusion:

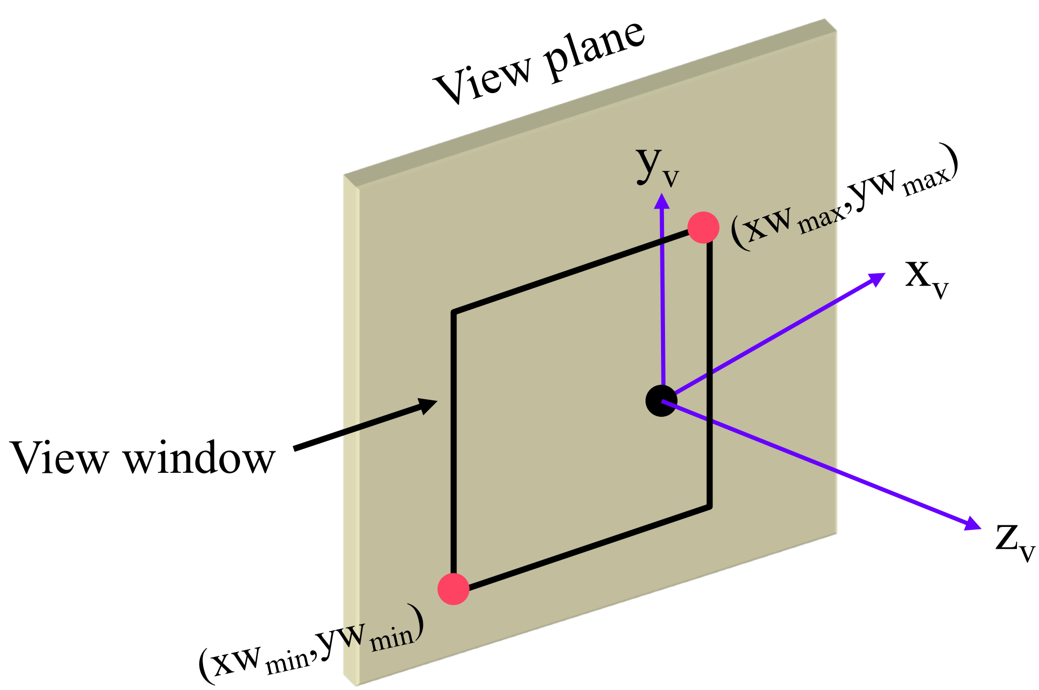
Both tabletops are the same size and shape



https://www.youtube.com/watch?v
=PhRbsaSMMVU&t=29s

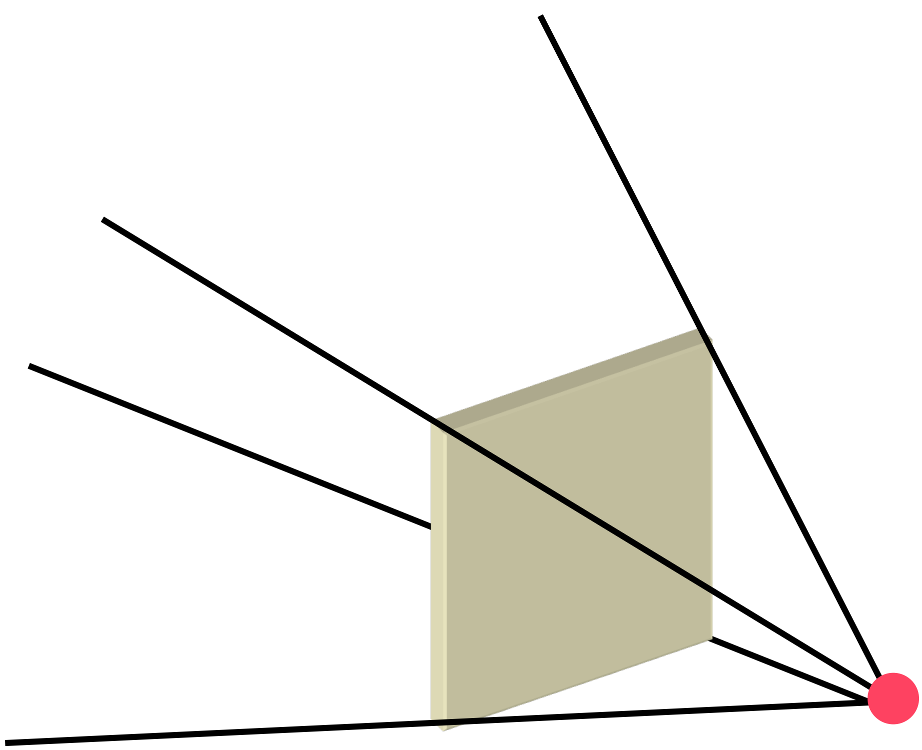
View Window

- After objects were projected onto the viewing plane, an image is taken from a *View Window*.
- A view window can be placed *anywhere* on the view plane.
- In general the view window is aligned with the viewing coordinates and is defined by its extreme points: (xw_{min},yw_{min}) and (xw_{max},yw_{max})

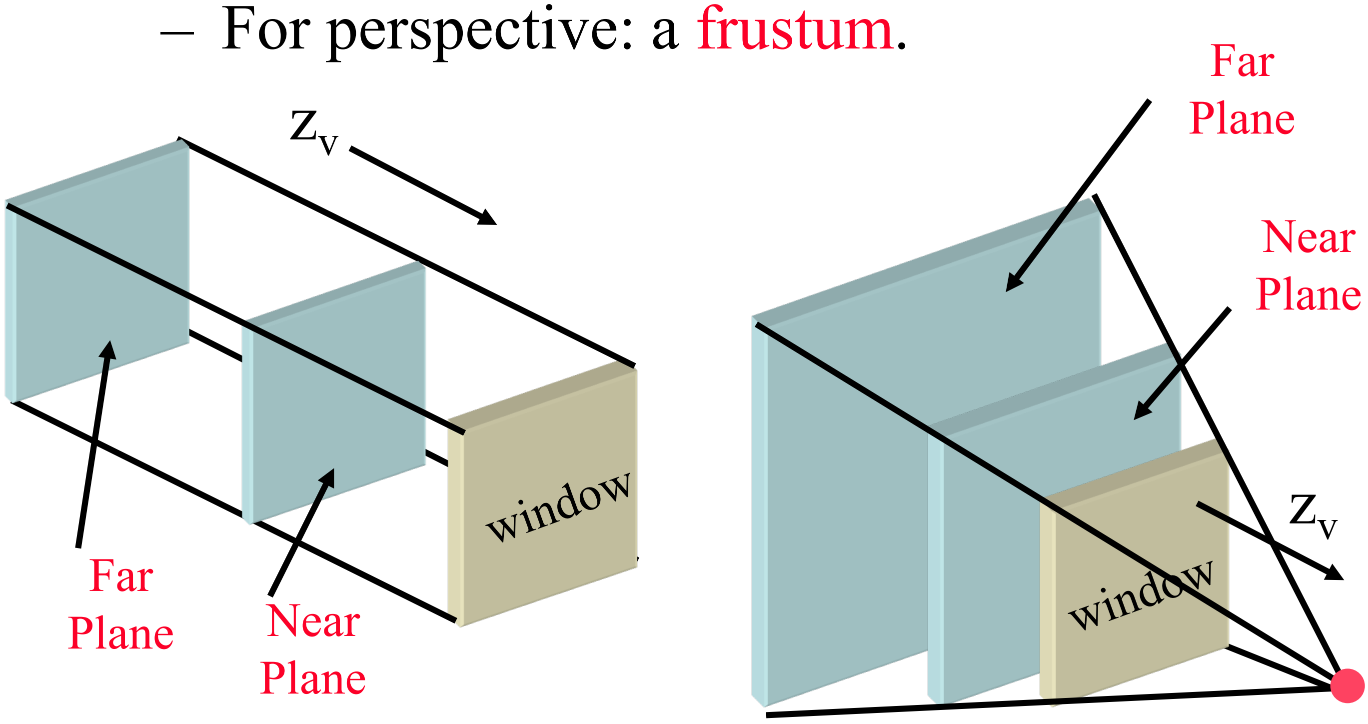


View Volume

- Given the specification of the *view window,* we can set up a *View Volume*.
- Only objects inside the view volume might appear in the display, the rest are clipped.

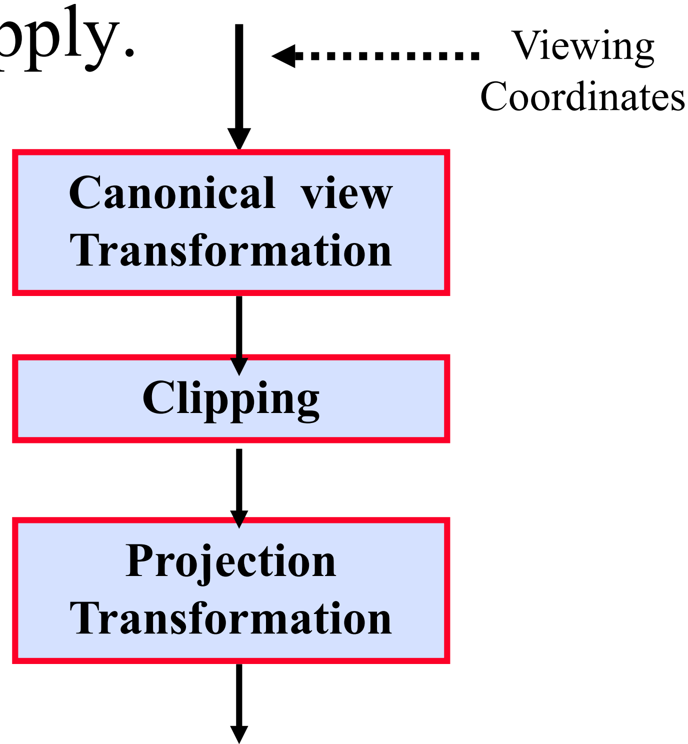


- In order to limit the infinite view volume we define two additional planes: *Near Plane* and *Far Plane*.
- Only objects in the bounded view volume can appear.
- The near and far planes are parallel to the view plane and specified by z_{near} and z_{far}.
- A limited view volume is defined:
  - For orthographic: a rectangular parallelpiped.
  - For oblique: an oblique parallelpiped.
  - For perspective: a frustum.

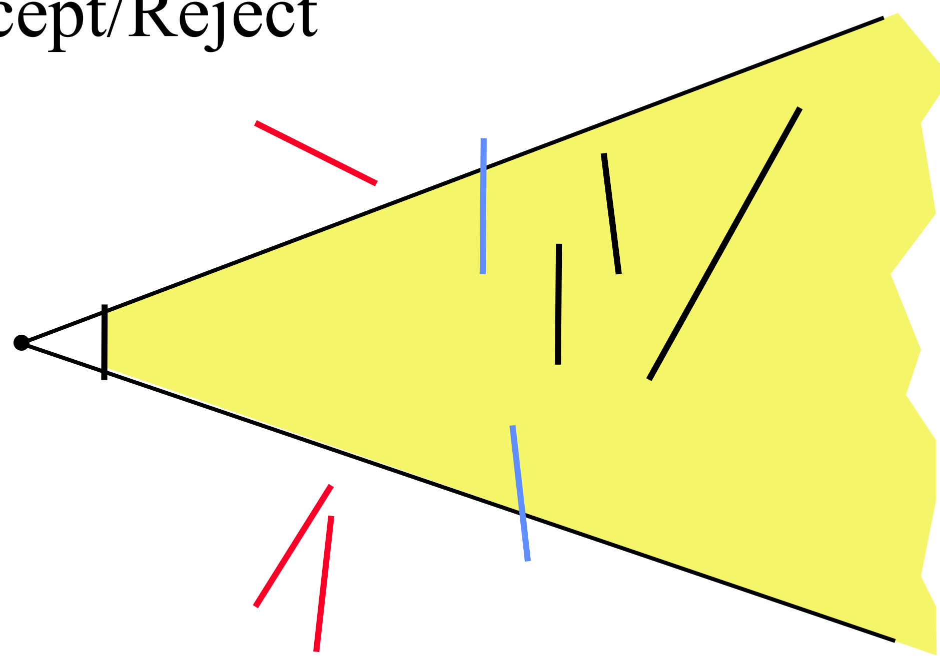


Canonical View Volumes

- In order to determine the objects that are seen in the view window we have to clip objects against six planes forming the view volume.
- Clipping against arbitrary 3D plane requires considerable computations.
- For fast clipping we transform the general view volume to a *canonical view volume* against which clipping is easy to apply.



Trivial Accept/Reject

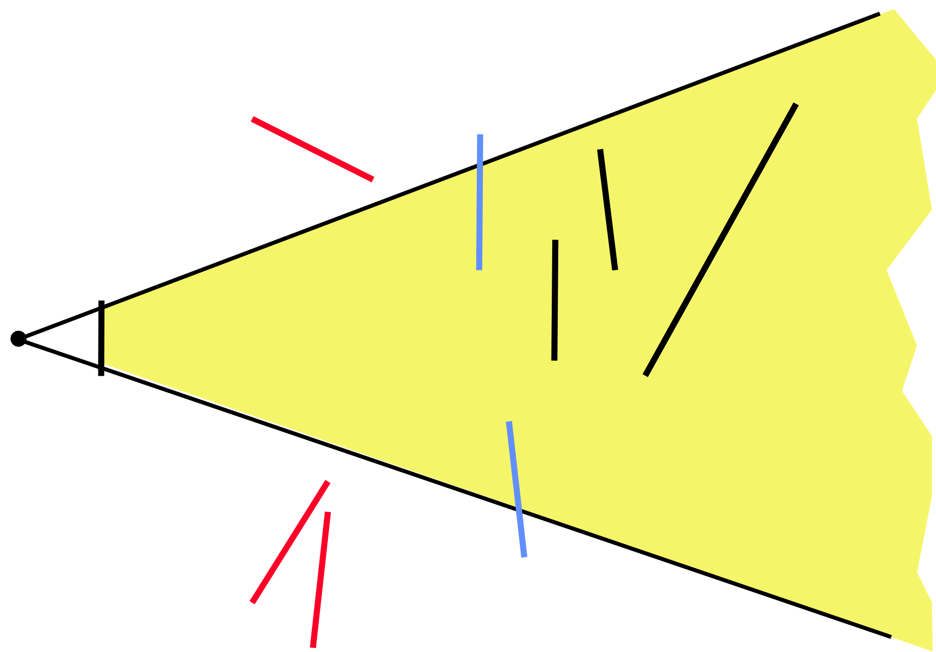


Red polygons are rejected

Black are accepted

And

Blue are tagged for clipping

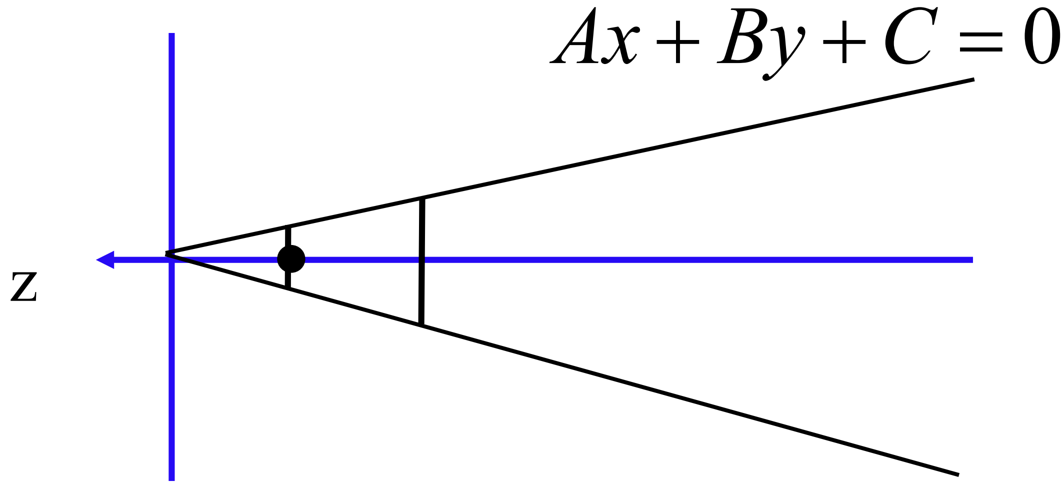


Classify the vertices of the polygons against each side Si of the frustum in turn:

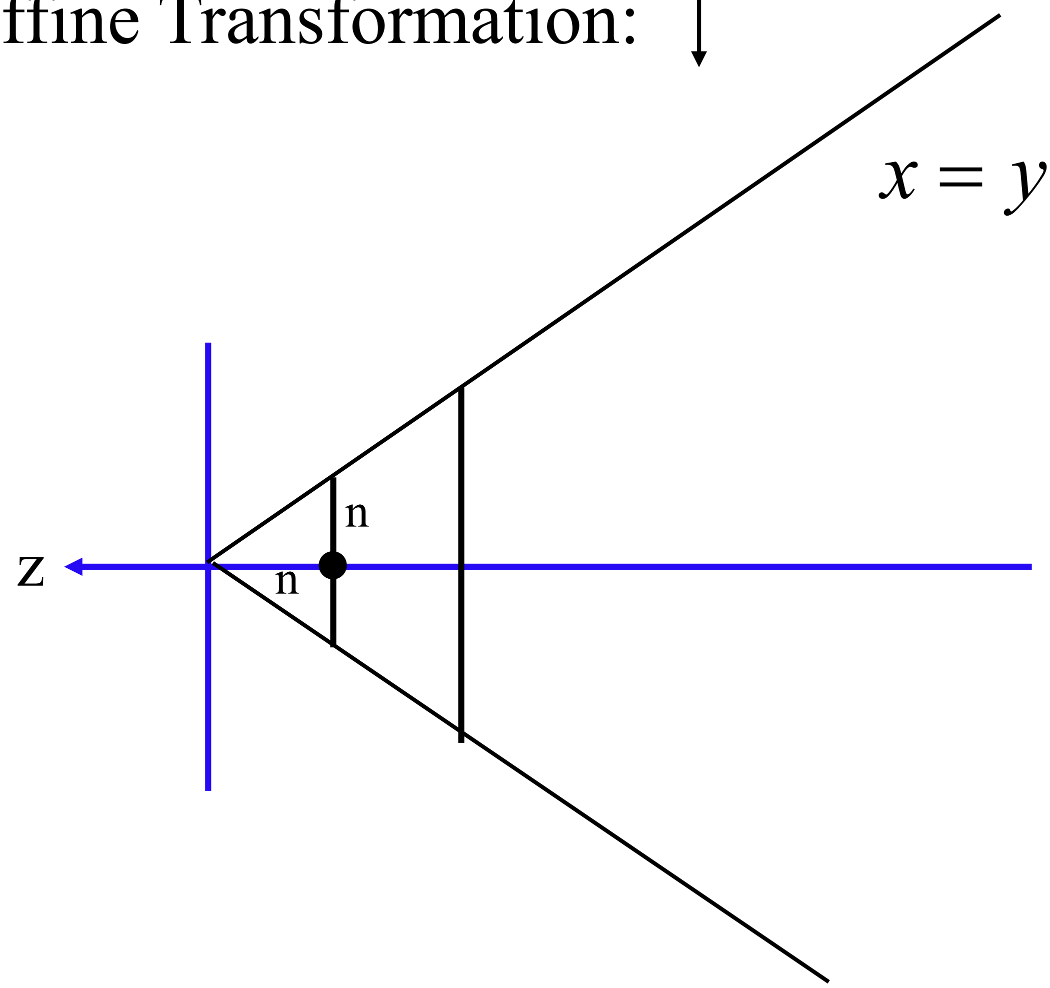
If all the vertices are on the outside of some Si, then cull the polygon, otherwise,

Polygons with all its vertices inside all Sj are accepted.

All others are tagged.



Affine Transformation:



OpenGL Transformation Pipe-Line