CMP205: Computer Graphics

CAIRO UNIVERSITY

Lecture 5: Viewing and Projection

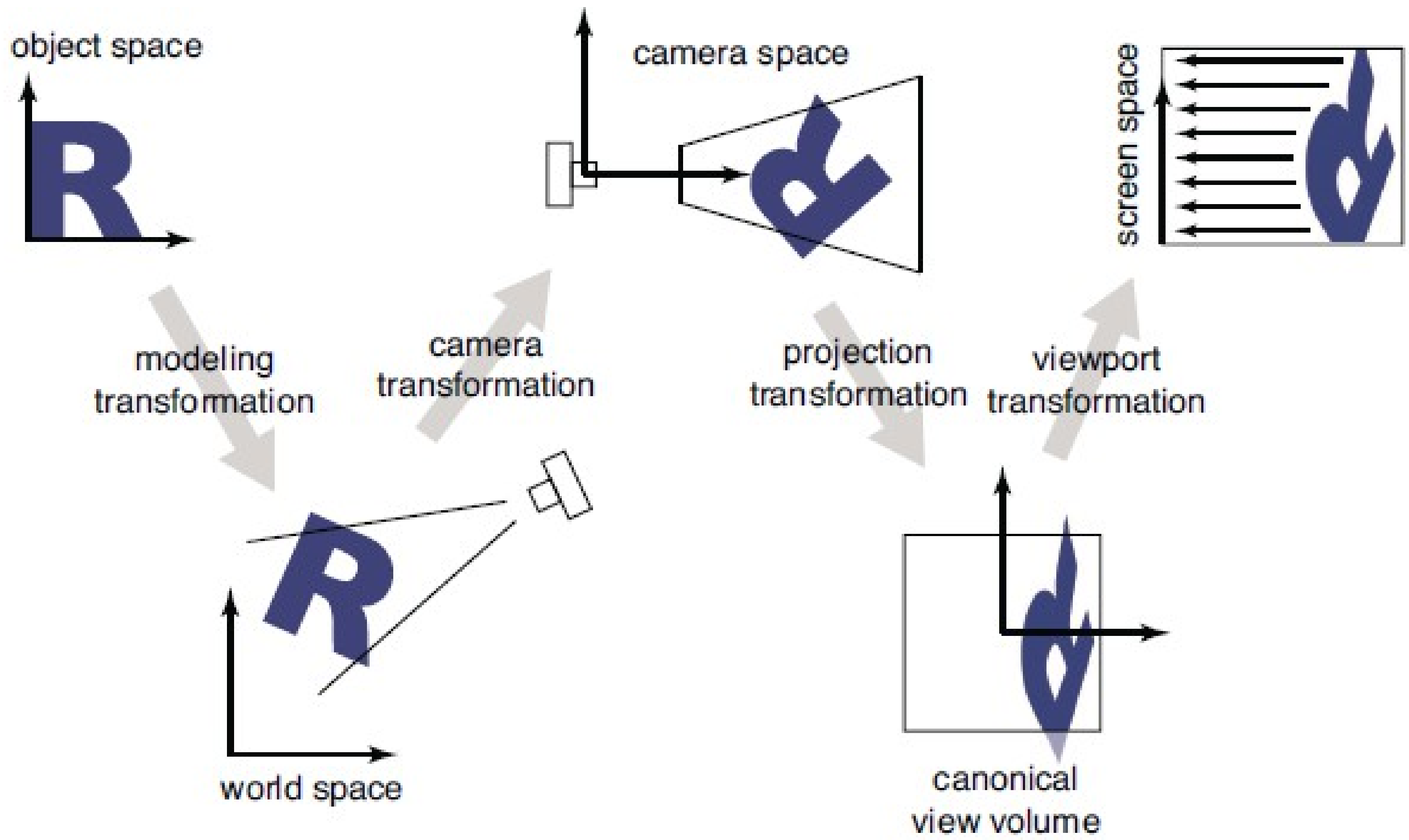
Mohamed Alaa El-Dien Aly
Computer Engineering Department
Cairo University
Fall 2012

Agenda

- Viewing
- Projections
  - Orthographic
  - Perspective
- Transformations Pipeline

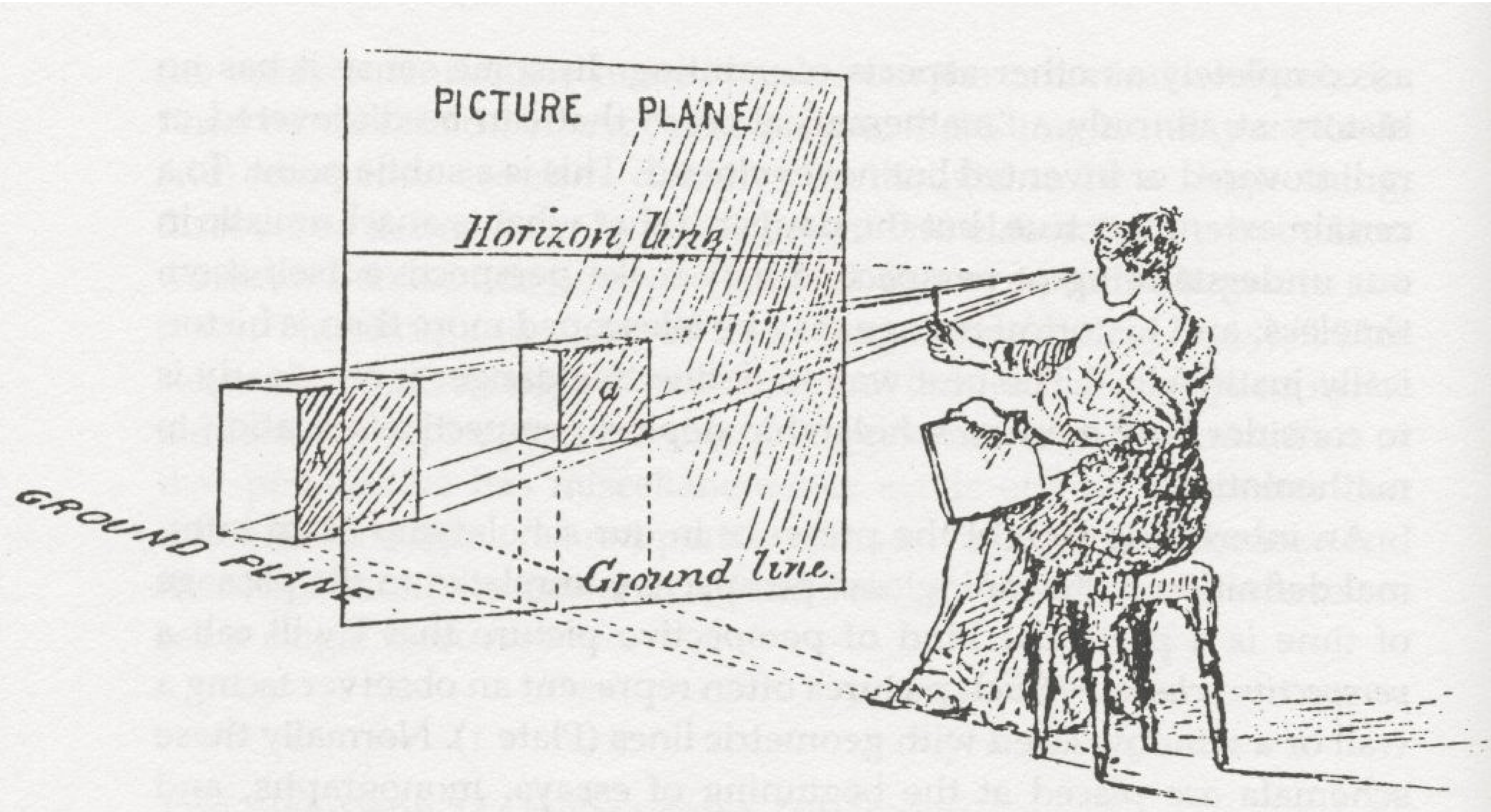
Acknowledgment: Some slides adapted from Steve Marschner and Maneesh Agrawala

3D Viewing



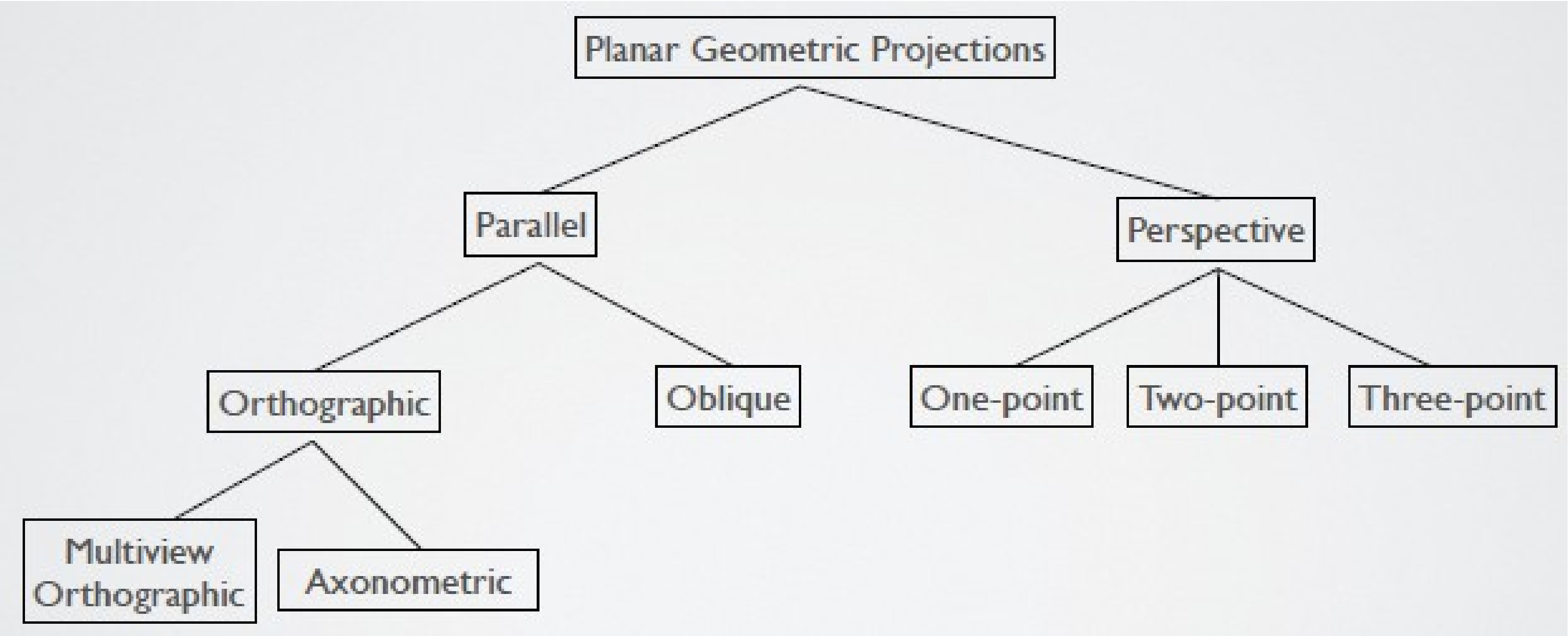
Convert from 3D points in space to 2D points on screen

Projections

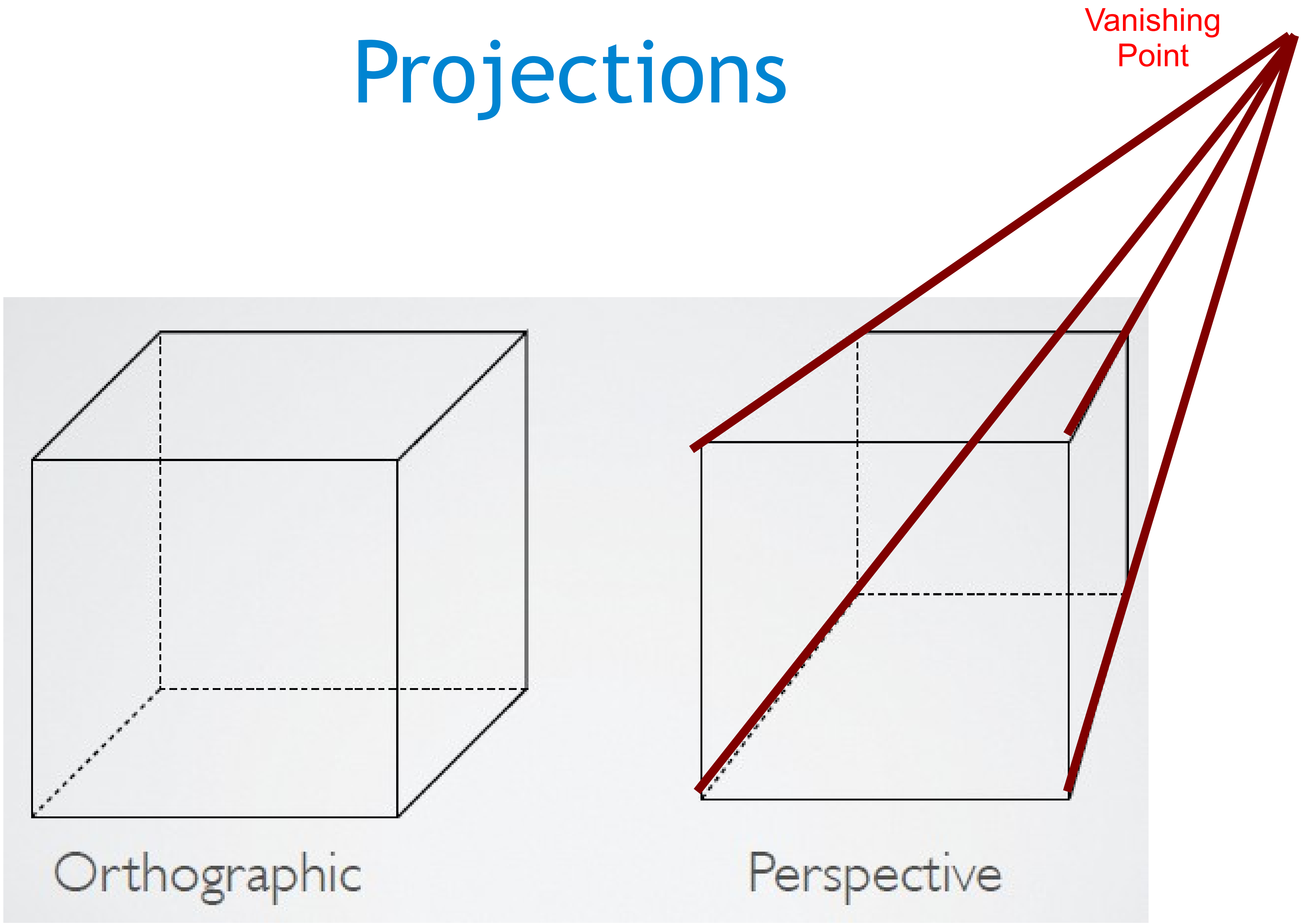


Project points in the world onto a plane

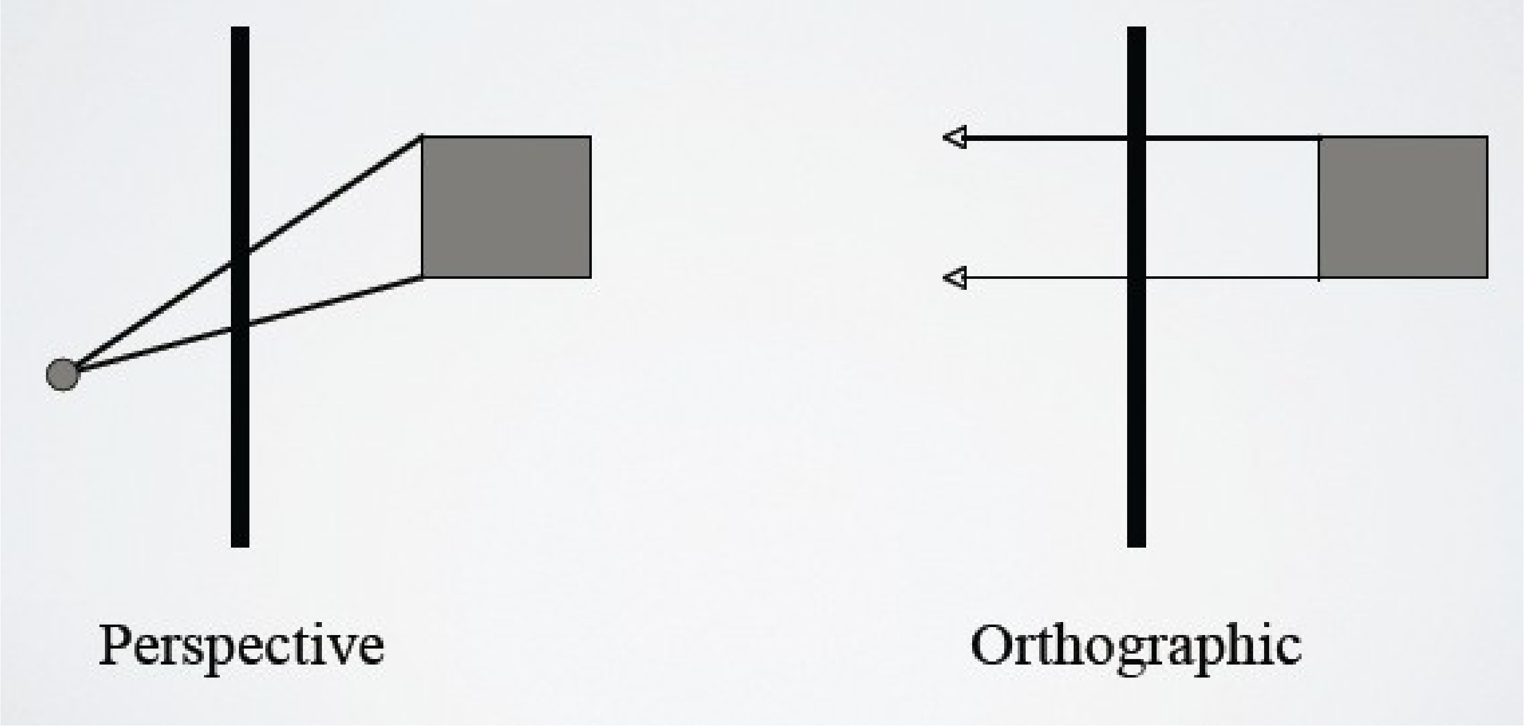
Projections



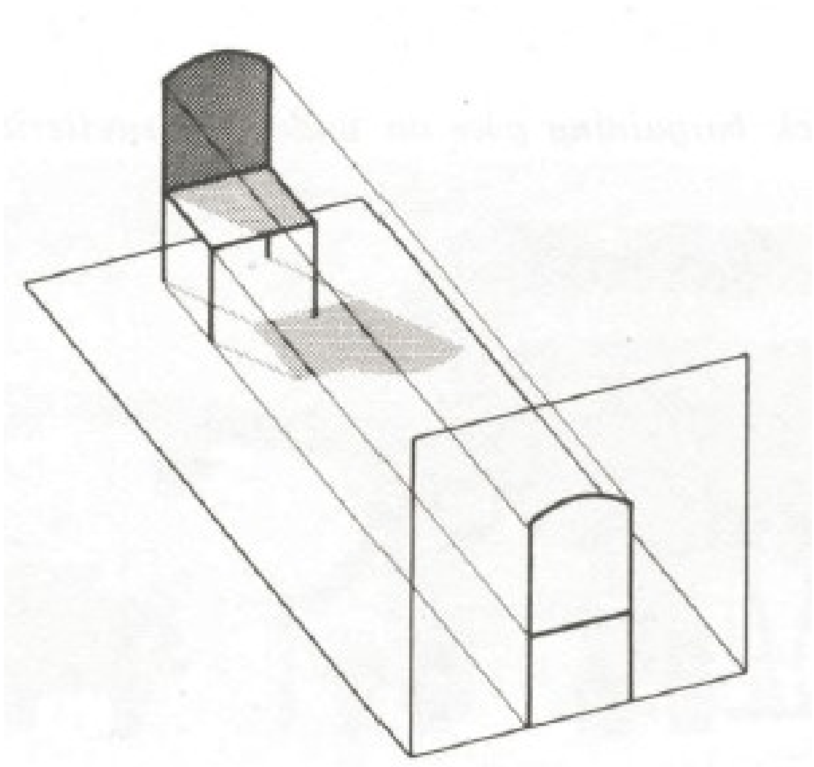
Projections



Projections in 2D

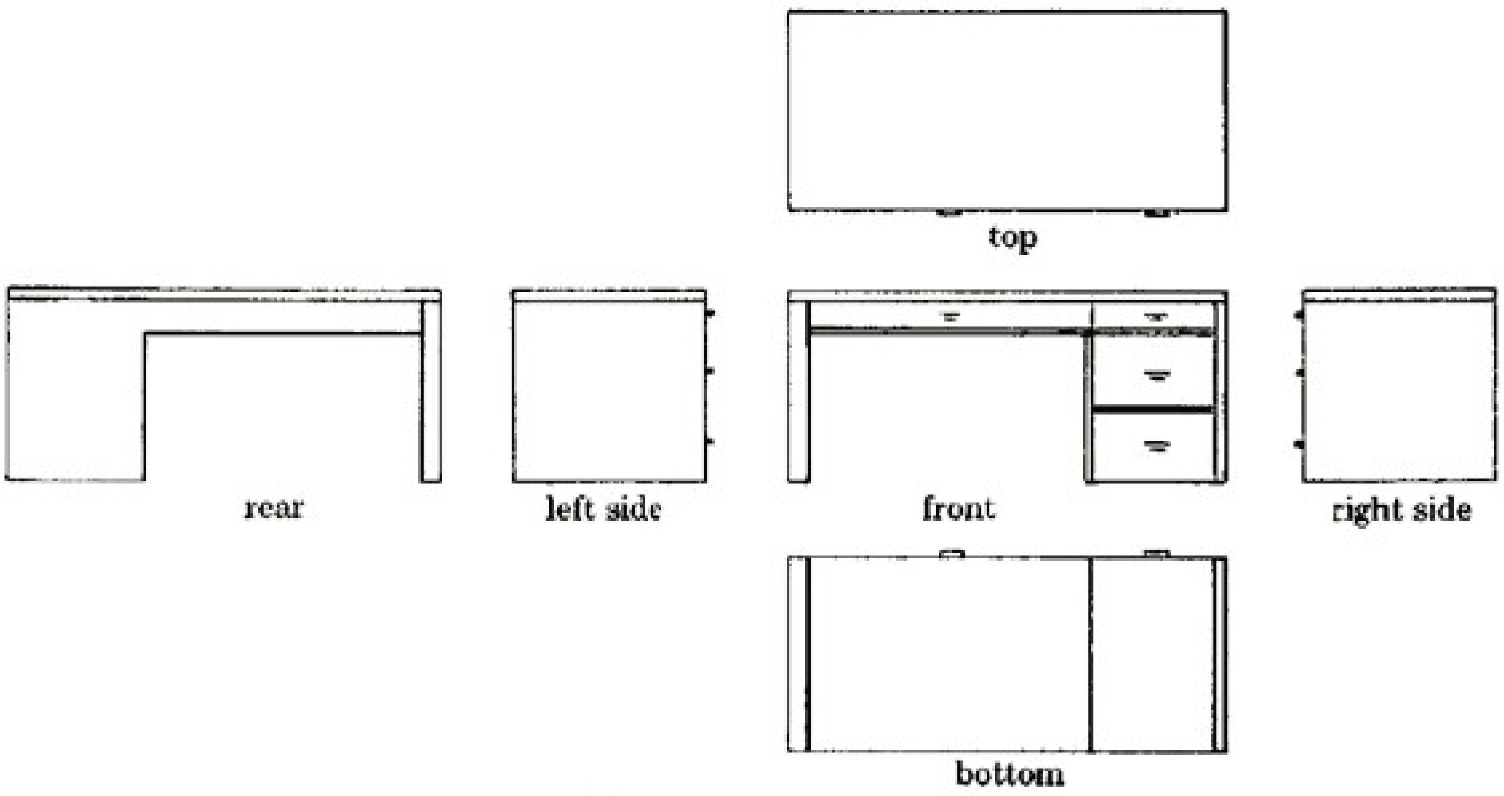


Orthographic Projection

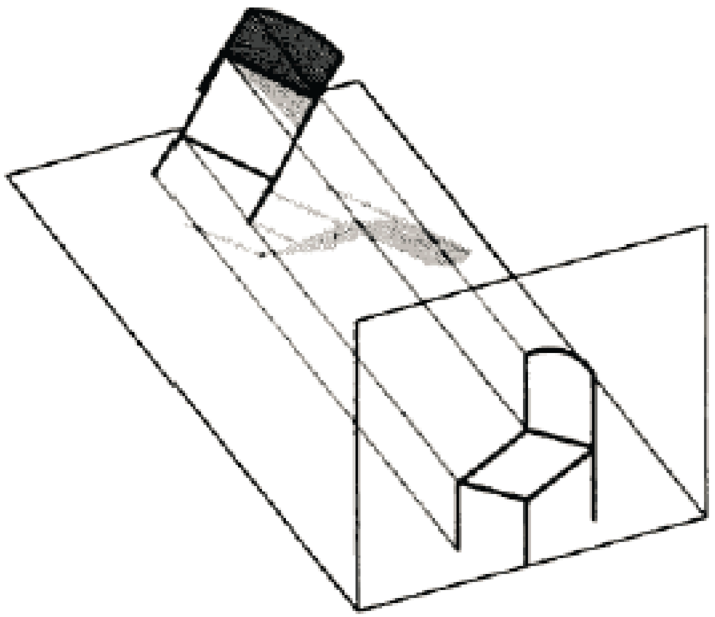


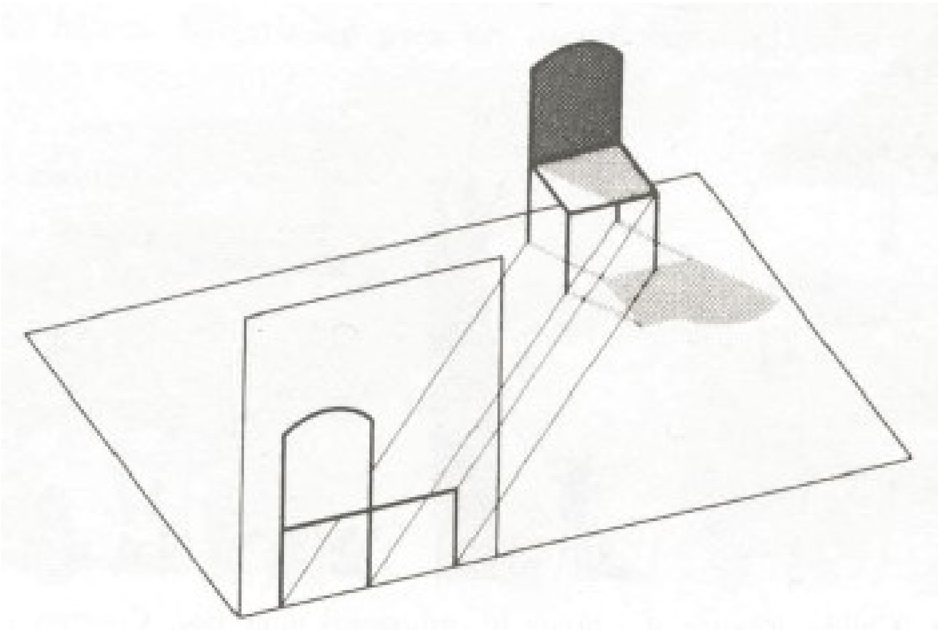
- Projection plane parallel to a Coordinate plane
- Projection direction perpendicular to projection plane

Multiview Orthographic



Off-Axis Projections





Axonometric Projection

Projection plane not parallel to coordinate planes

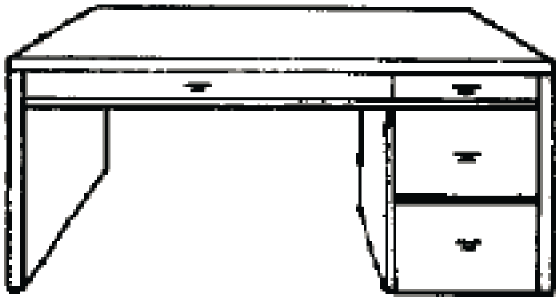
Oblique Projection

Projection lines not perpendicular to projection plane

Perspective Projection

One-Point Perspective

Projection plane parallel to
a coordinate plane



one-point

Two-Point Perspective

Projection plane parallel to
a coordinate axis

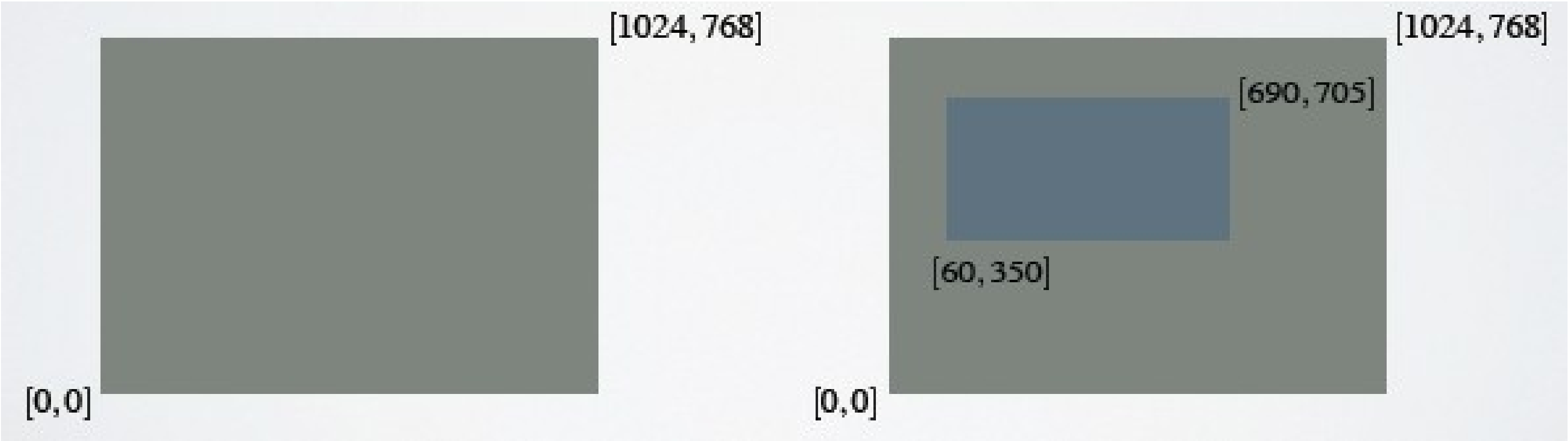
two-point

Three-Point Perspective

Projection plane not parallel to
any coordinate axes

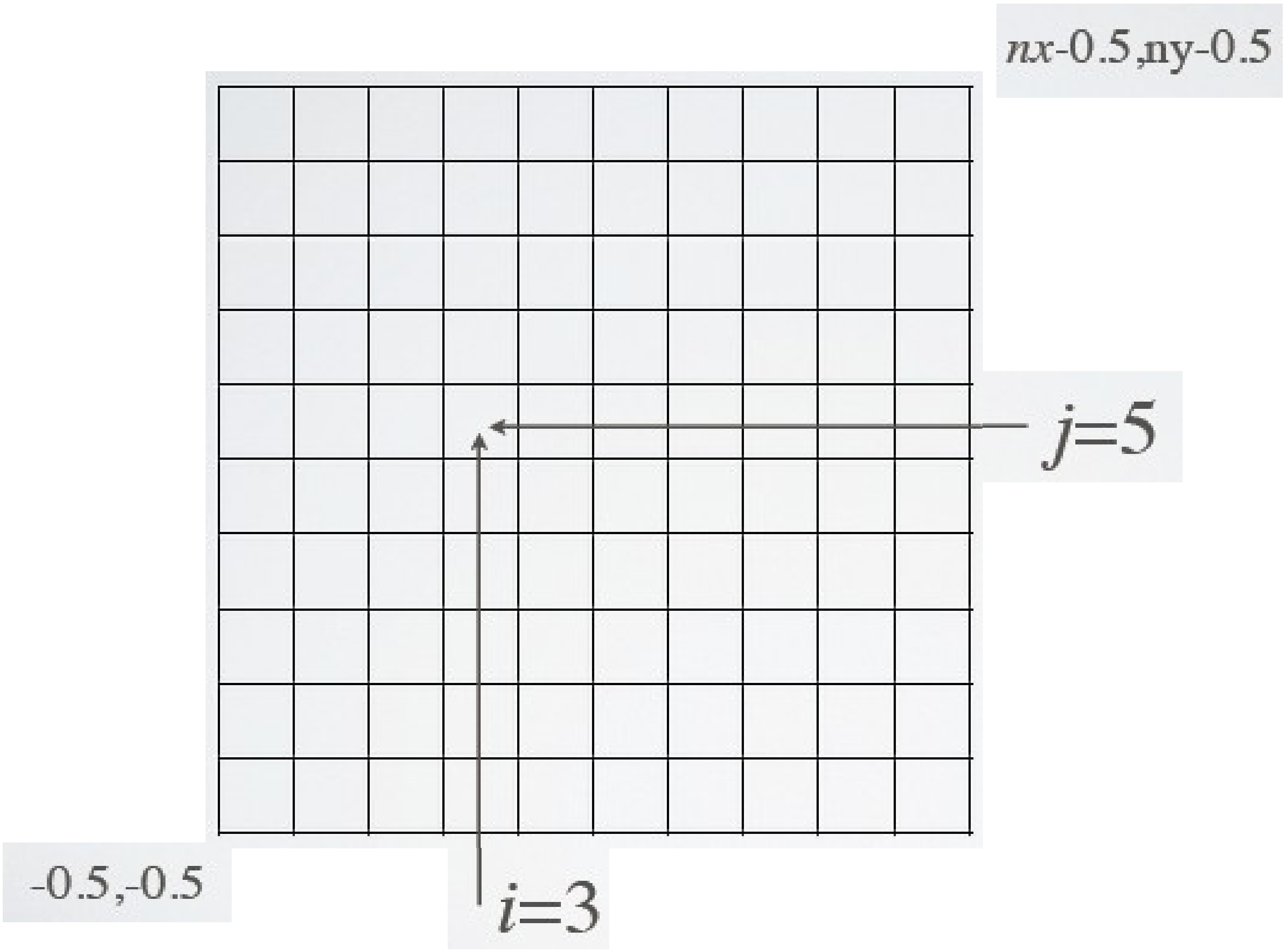
three-point

Screen Space



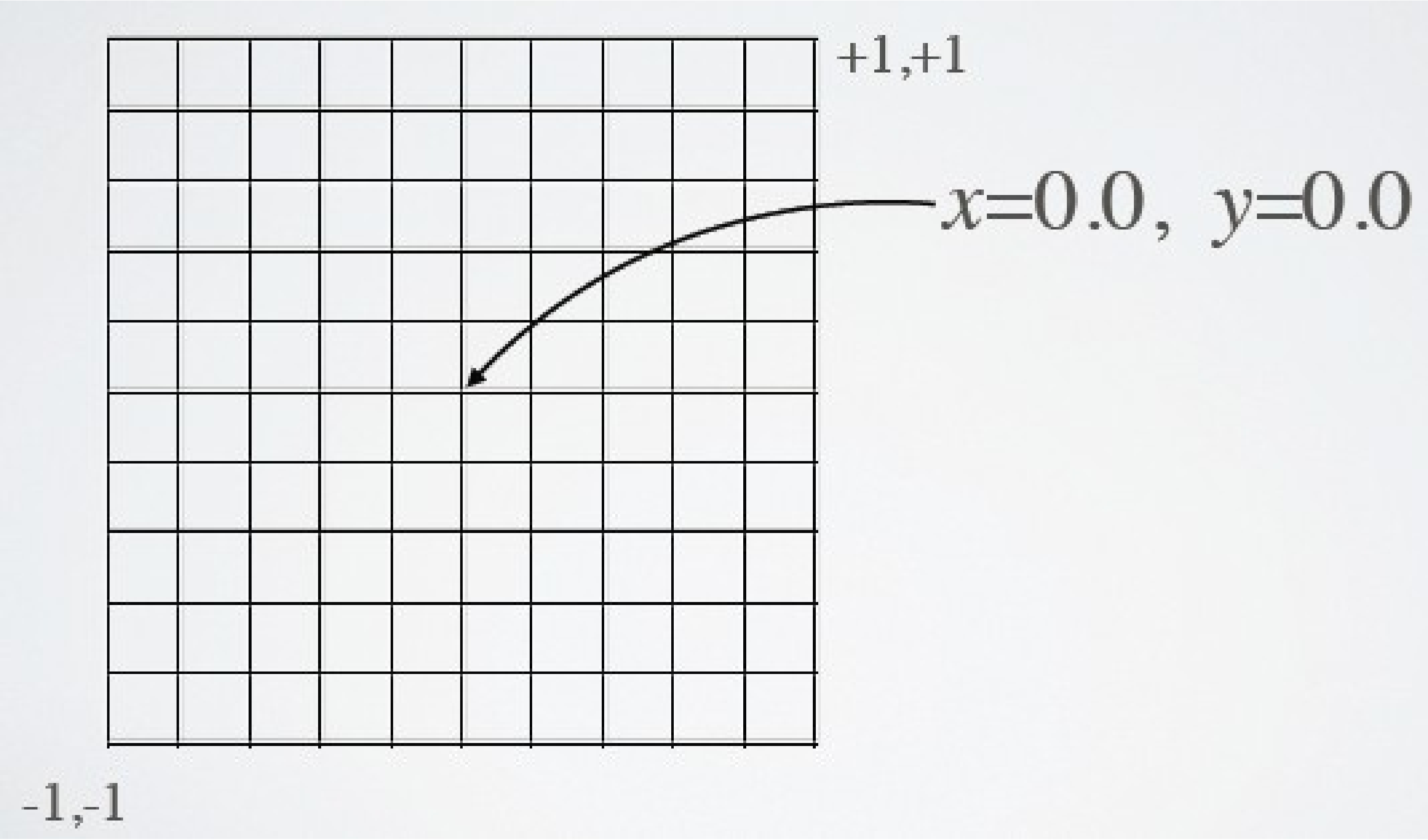
Viewport

Screen Space

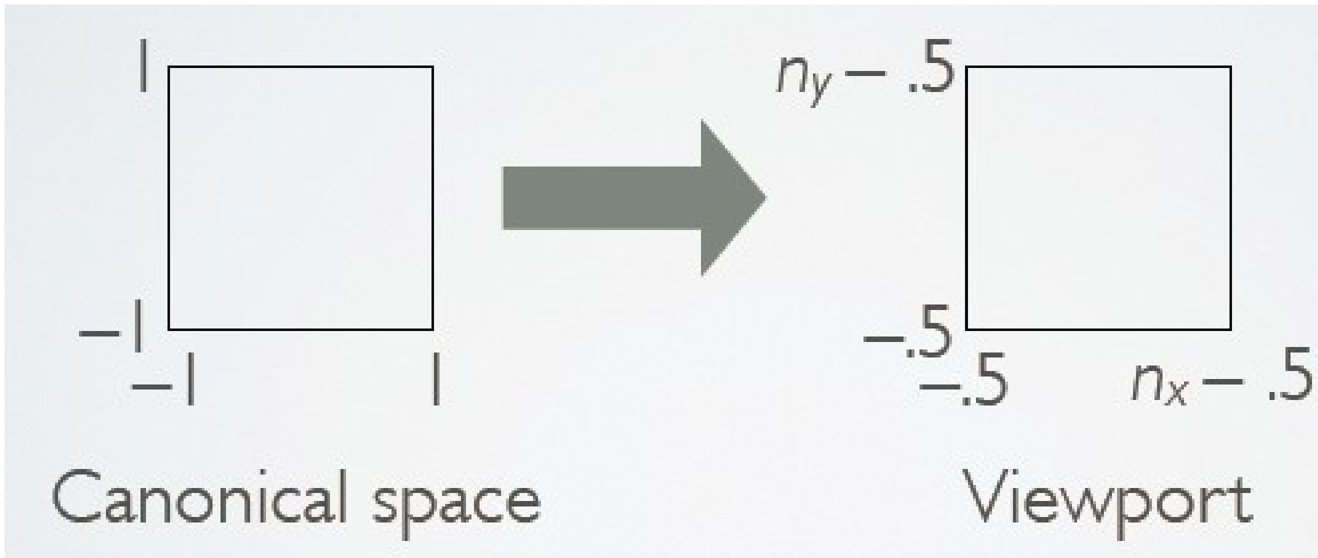


Screen of width n_x and height n_y pixels

2D Canonical View Space

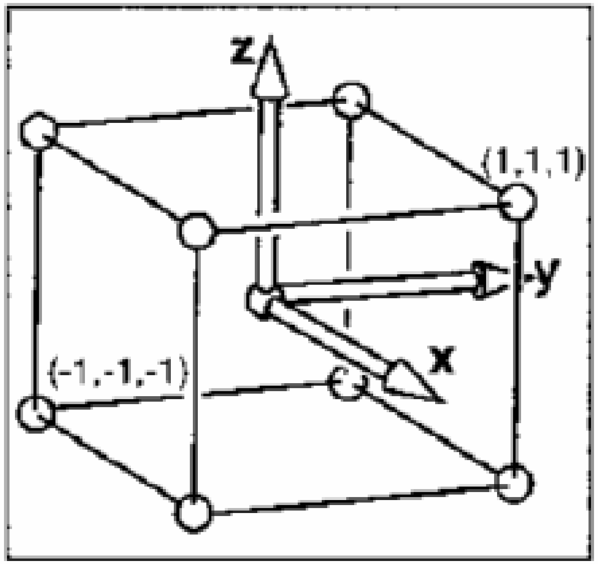


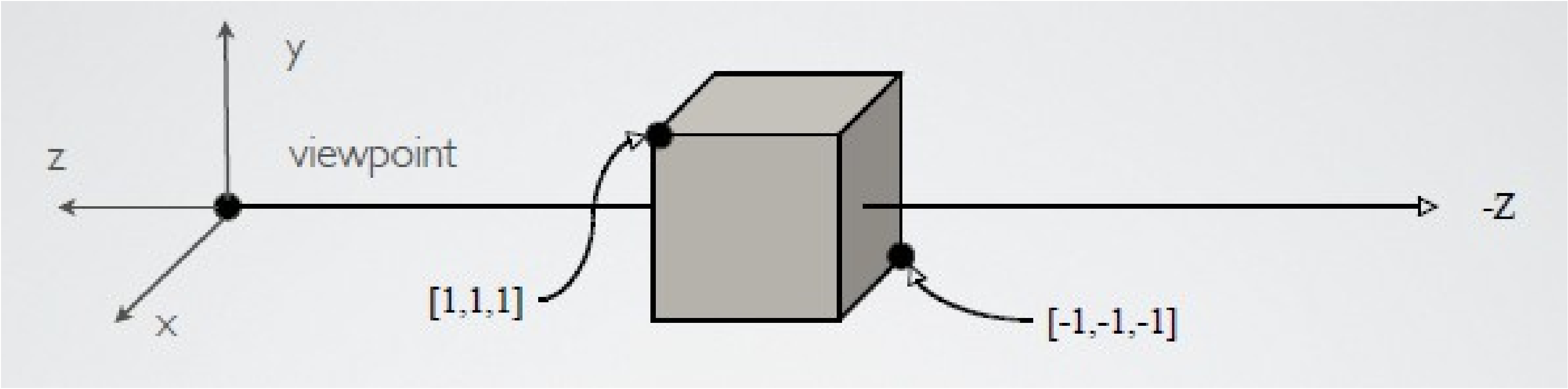
2D Viewport Transformation



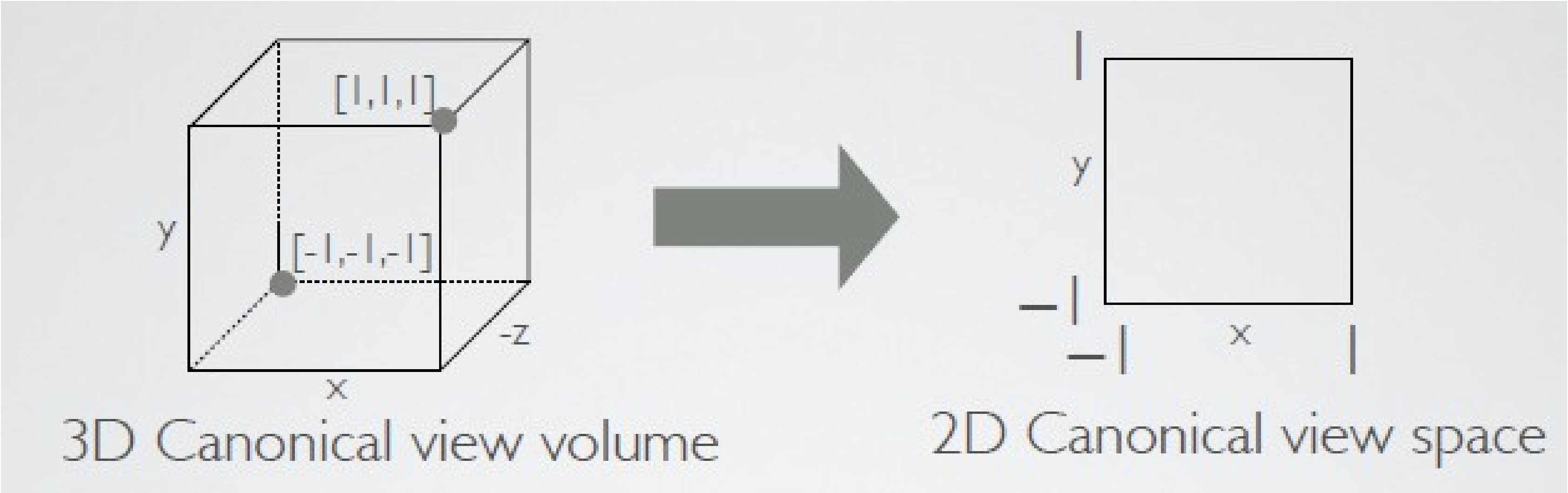
$$\begin{bmatrix} x_{screen} \\ y_{screen} \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{n_x}{2} & 0 & \frac{n_x - 1}{2} \\ 0 & \frac{n_y}{2} & \frac{n_y - 1}{2} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_{canonical} \\ y_{canonical} \\ 1 \end{bmatrix}$$

3D Canonical View Volume





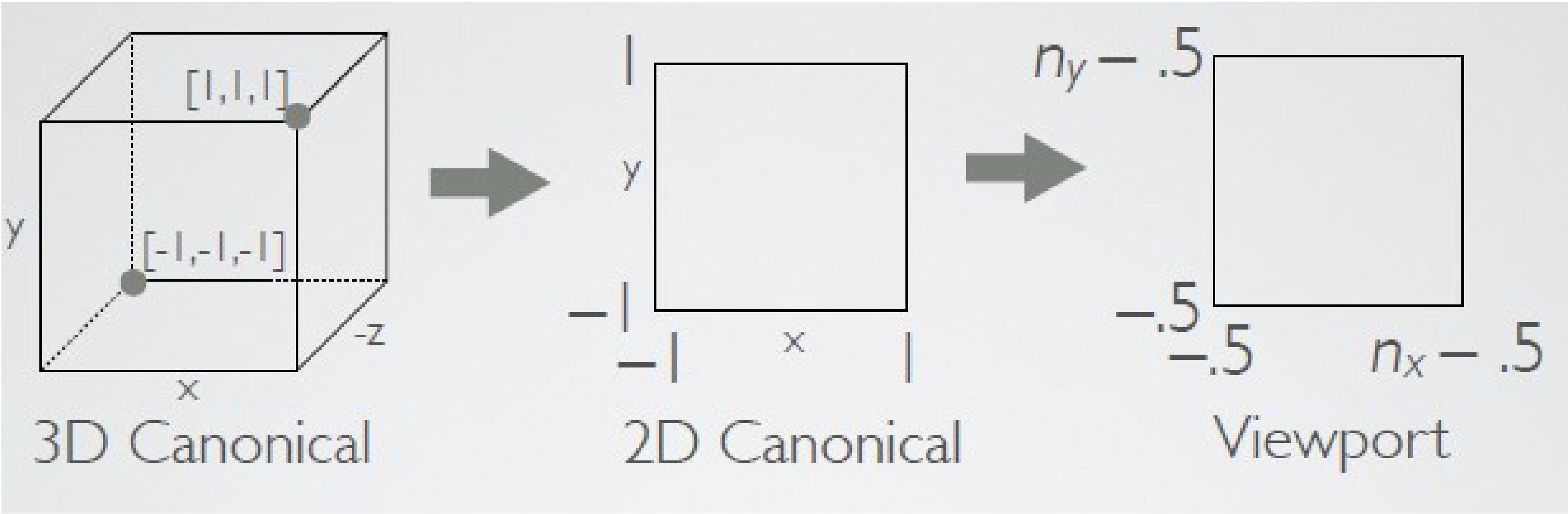
3D Viewport Transformation



$$\begin{bmatrix} x_{canonical} \\ y_{canonical} \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_{canonical} \\ y_{canonical} \\ z_{canonical} \\ 1 \end{bmatrix}$$

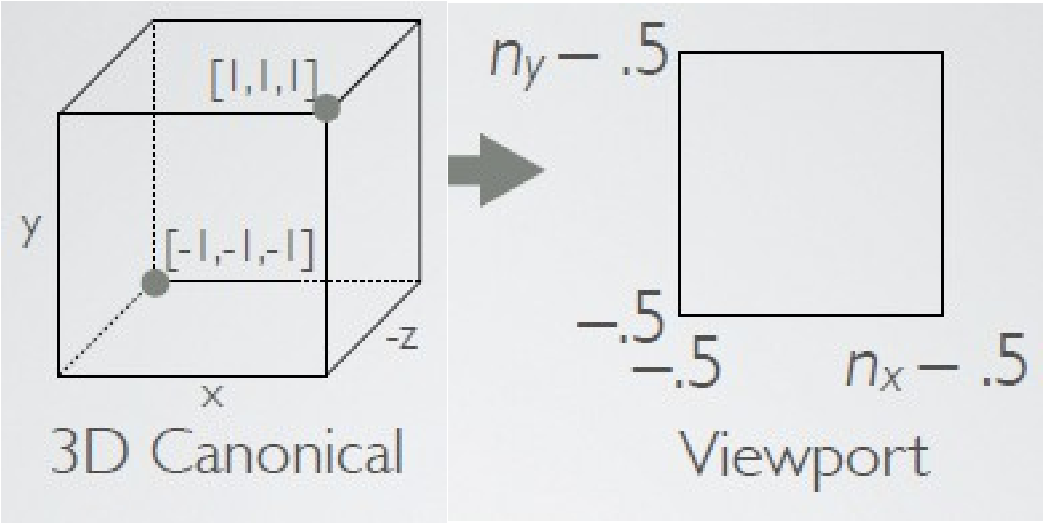
Drop z-coordinate

3D Viewport Transformation



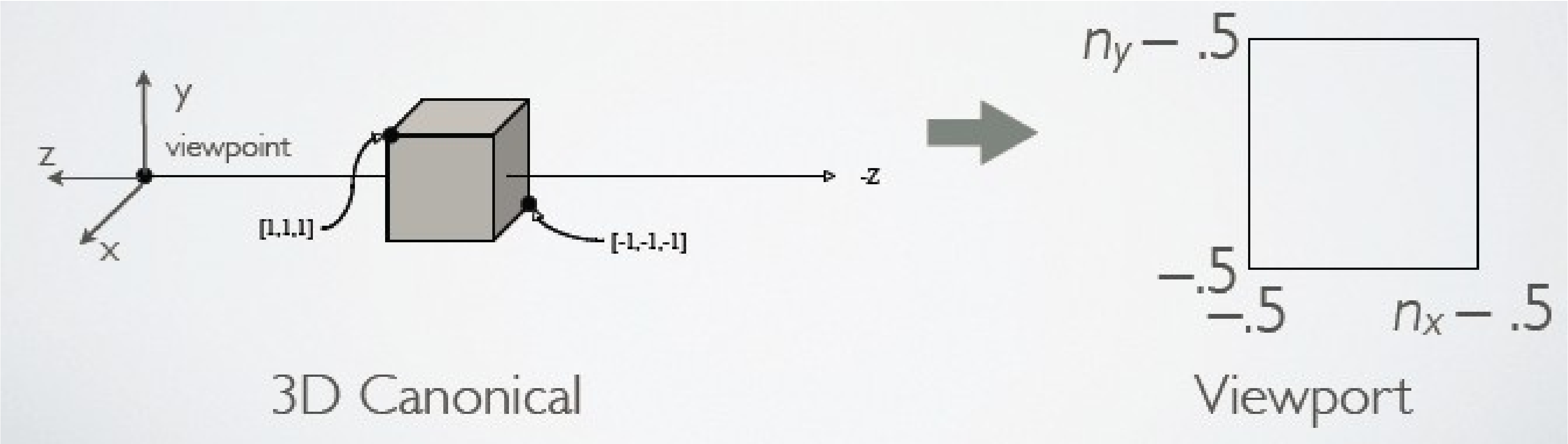
$$\begin{bmatrix} x_{screen} \\ y_{screen} \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{n_x}{2} & 0 & \frac{n_x - 1}{2} \\ 0 & \frac{n_y}{2} & \frac{n_y - 1}{2} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_{canonical} \\ y_{canonical} \\ z_{canonical} \\ 1 \end{bmatrix}$$

3D Viewport Transformation



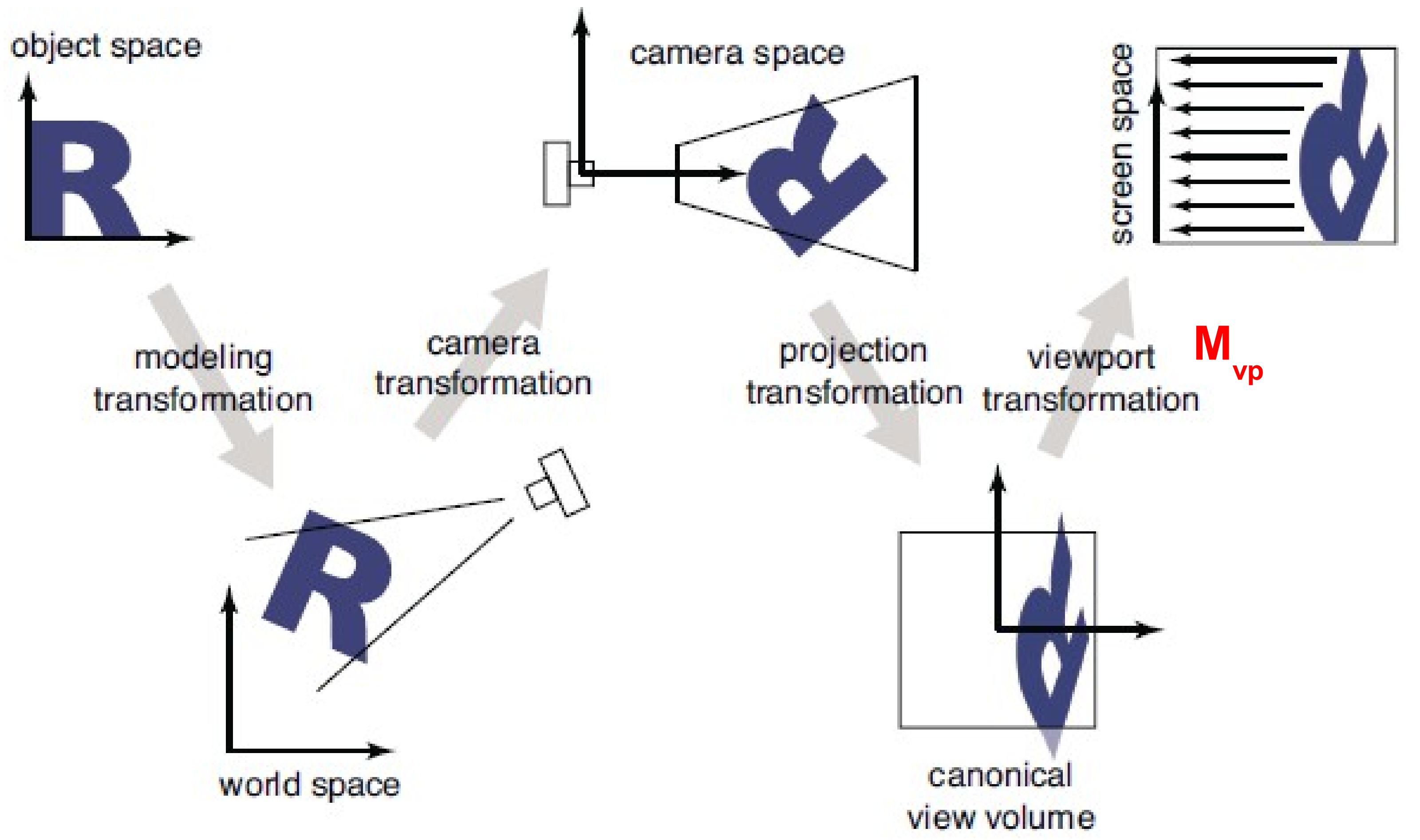
$$\begin{bmatrix} x_{screen} \\ y_{screen} \\ z_{canonical} \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{n_x}{2} & 0 & 0 & \frac{n_x - 1}{2} \\ 0 & \frac{n_y}{2} & 0 & \frac{n_y - 1}{2} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_{canonical} \\ y_{canonical} \\ z_{canonical} \\ 1 \end{bmatrix}$$

3D Viewport Transformation

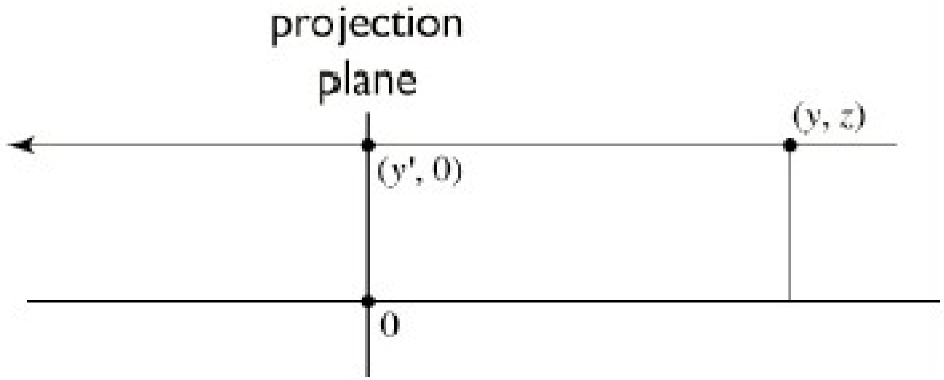


$$\boldsymbol{M}_{vp} = \begin{bmatrix} \frac{n_x}{2} & 0 & 0 & \frac{n_x - 1}{2} \\ 0 & \frac{n_y}{2} & 0 & \frac{n_y - 1}{2} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

3D Viewport Transformation



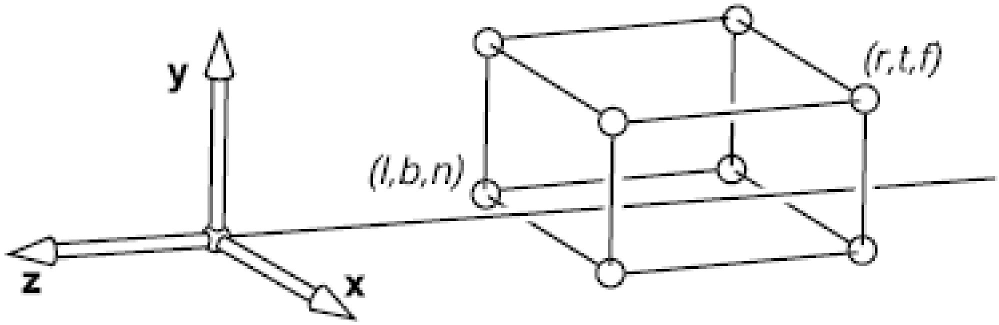
Orthographic Projection



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

Drop z-coordinate

Orthographic Projection

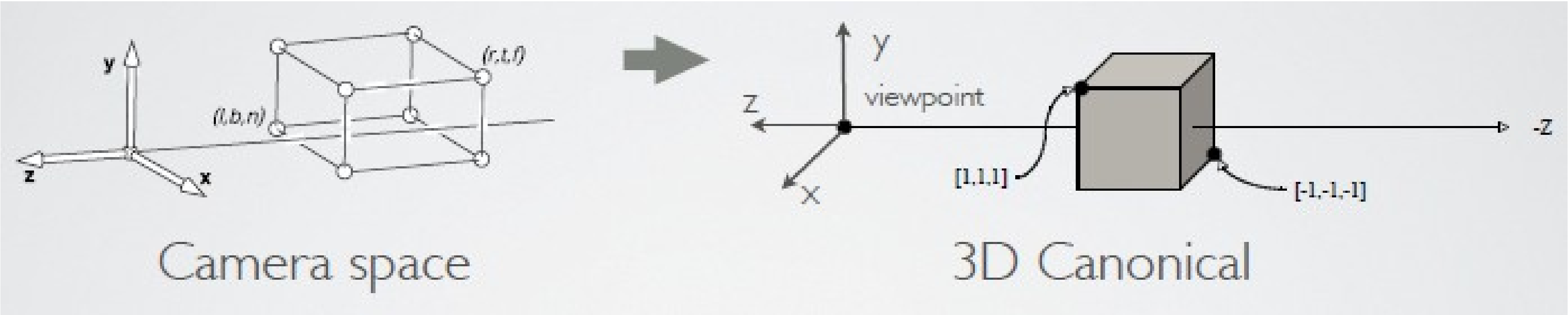


General View Volume

How do we transform this view volume to the canonical view volume?

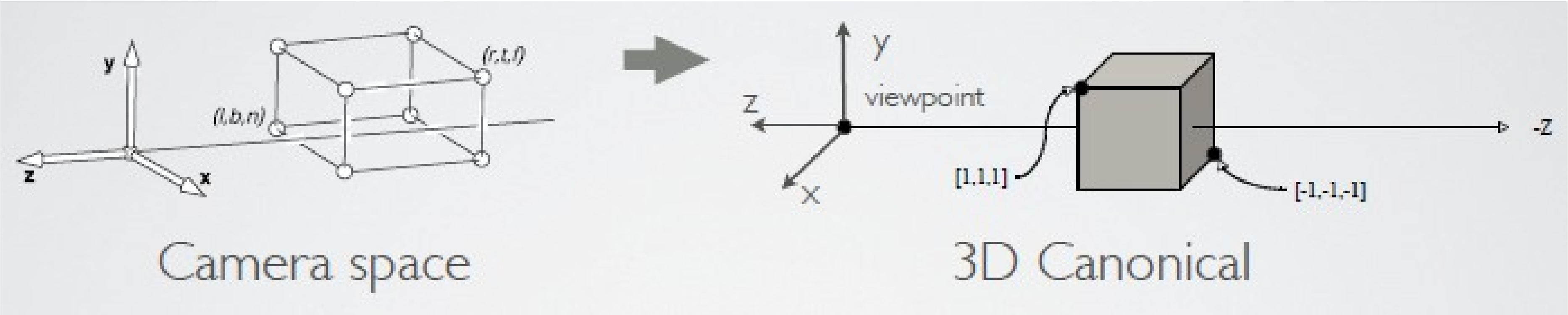
Windowing Transform !

Orthographic Projection



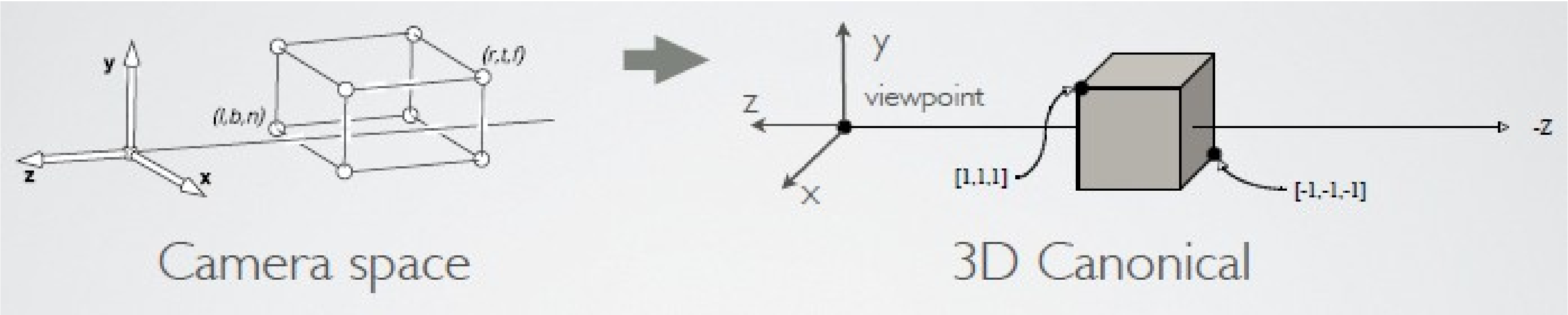
$$M_{orth} = \begin{bmatrix} \frac{2}{r-l} & 0 & 0 & 0 \\ 0 & \frac{2}{t-b} & 0 & 0 \\ 0 & 0 & \frac{2}{n-f} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & \frac{-(l+r)}{2} \\ 0 & 1 & 0 & \frac{-(b+t)}{2} \\ 0 & 0 & 1 & \frac{-(n+f)}{2} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Orthographic Projection



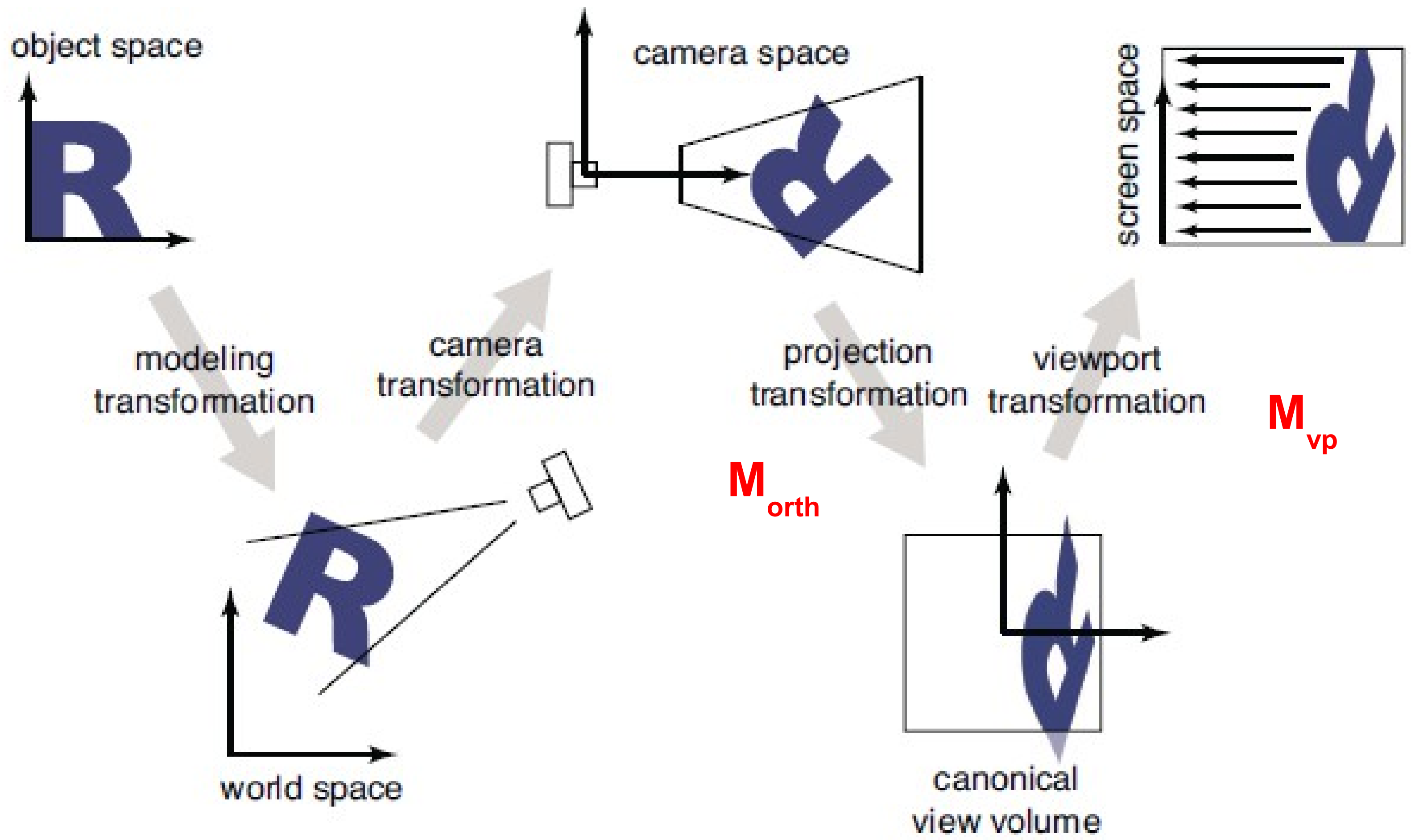
$$M_{orth} = \begin{bmatrix} \frac{2}{r-l} & 0 & 0 & \frac{-r+l}{r-l} \\ 0 & \frac{2}{t-b} & 0 & \frac{-t+b}{t-b} \\ 0 & 0 & \frac{2}{n-f} & \frac{-n+f}{n-f} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Orthographic Projection



$$\begin{bmatrix} x_{canonical} \\ y_{canonical} \\ z_{canonical} \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{2}{r-l} & 0 & 0 & \frac{-r+l}{r-l} \\ 0 & \frac{2}{t-b} & 0 & \frac{-t+b}{t-b} \\ 0 & 0 & \frac{2}{n-f} & \frac{-n+f}{n-f} \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_{cam} \\ y_{cam} \\ z_{cam} \\ 1 \end{bmatrix}$$

Orthographic Projection



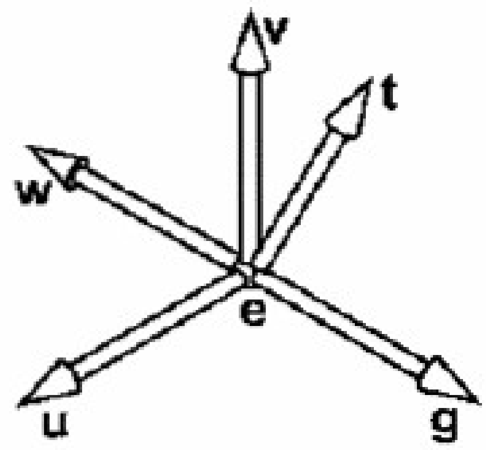
Arbitrary Views

Camera position/direction

e : eye position

g : gaze direction

t : view up vector



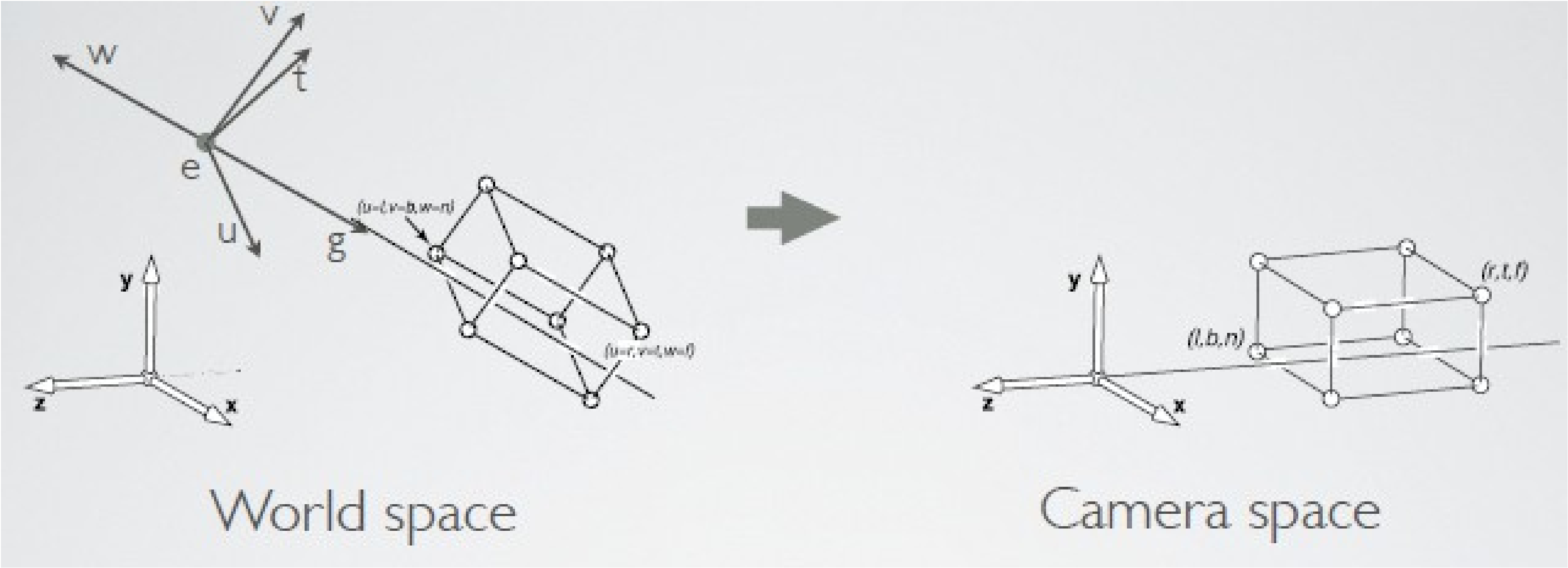
Construct a coordinate system

$$w = \frac{-g}{\|g\|}$$

$$u = \frac{-t \times w}{\|t \times w\|}$$

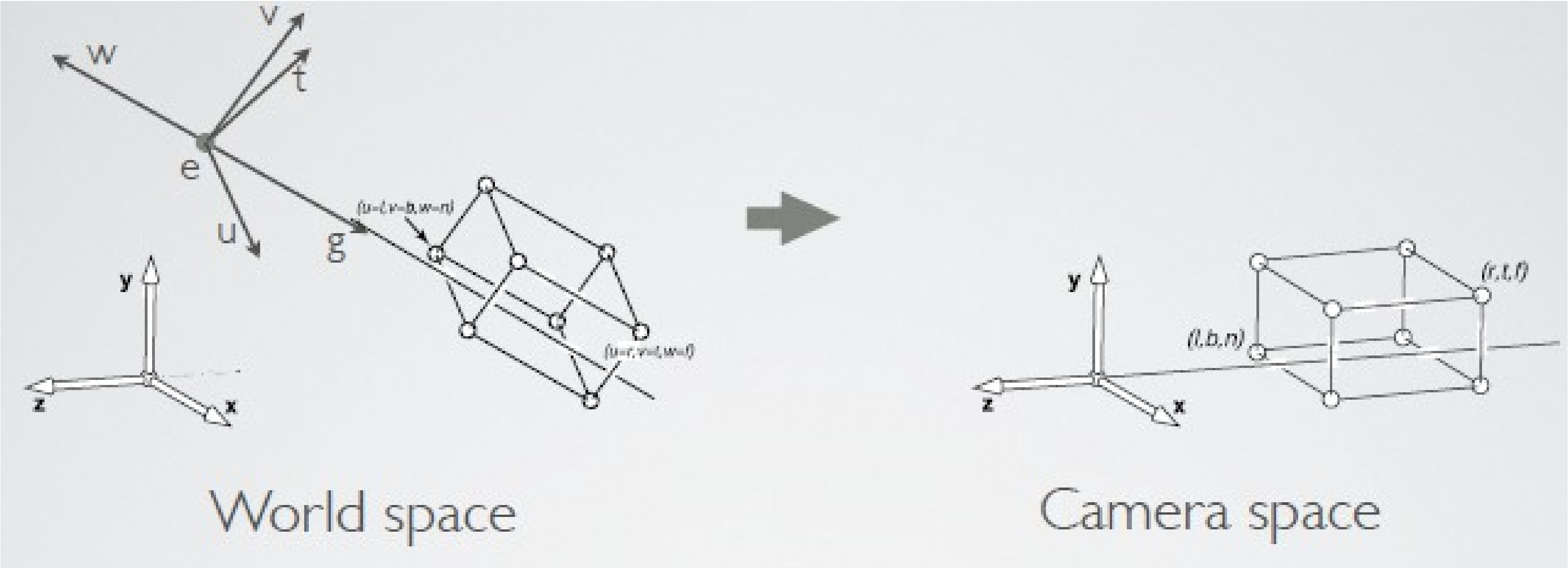
$$v = w \times u$$

Camera Transformation



Convert from World Coordinates to Camera Coordinates

Camera Transformation

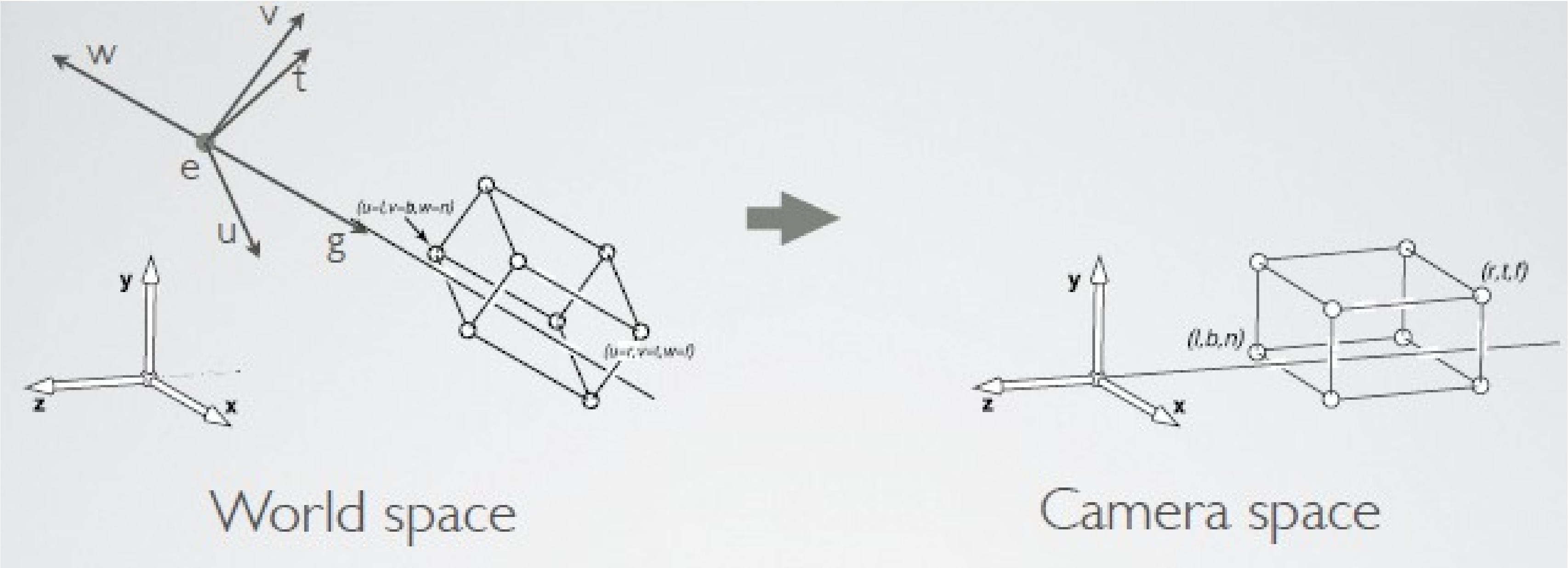


$$M_{cam} = \begin{bmatrix} x_u & y_u & z_u & 0 \\ x_v & y_v & z_v & 0 \\ x_w & y_w & z_w & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & -x_e \\ 0 & 1 & 0 & -y_e \\ 0 & 0 & 1 & -z_e \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} u & v & w & e \\ 0 & 0 & 0 & 1 \end{bmatrix}^{-1}$$

Aligns Camera Coordinates with World Coordinates

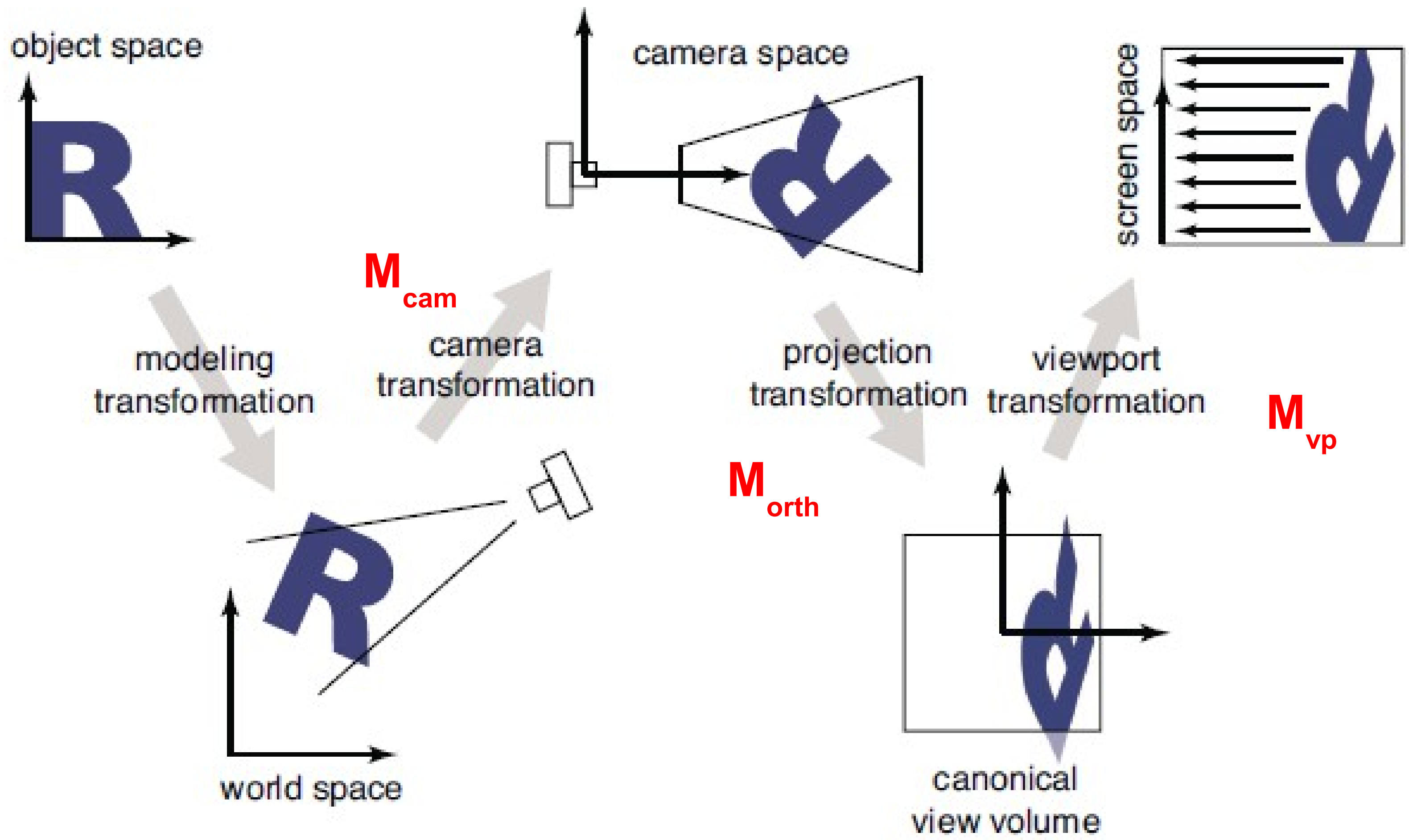
Moves Camera to World Origin

Camera Transformation

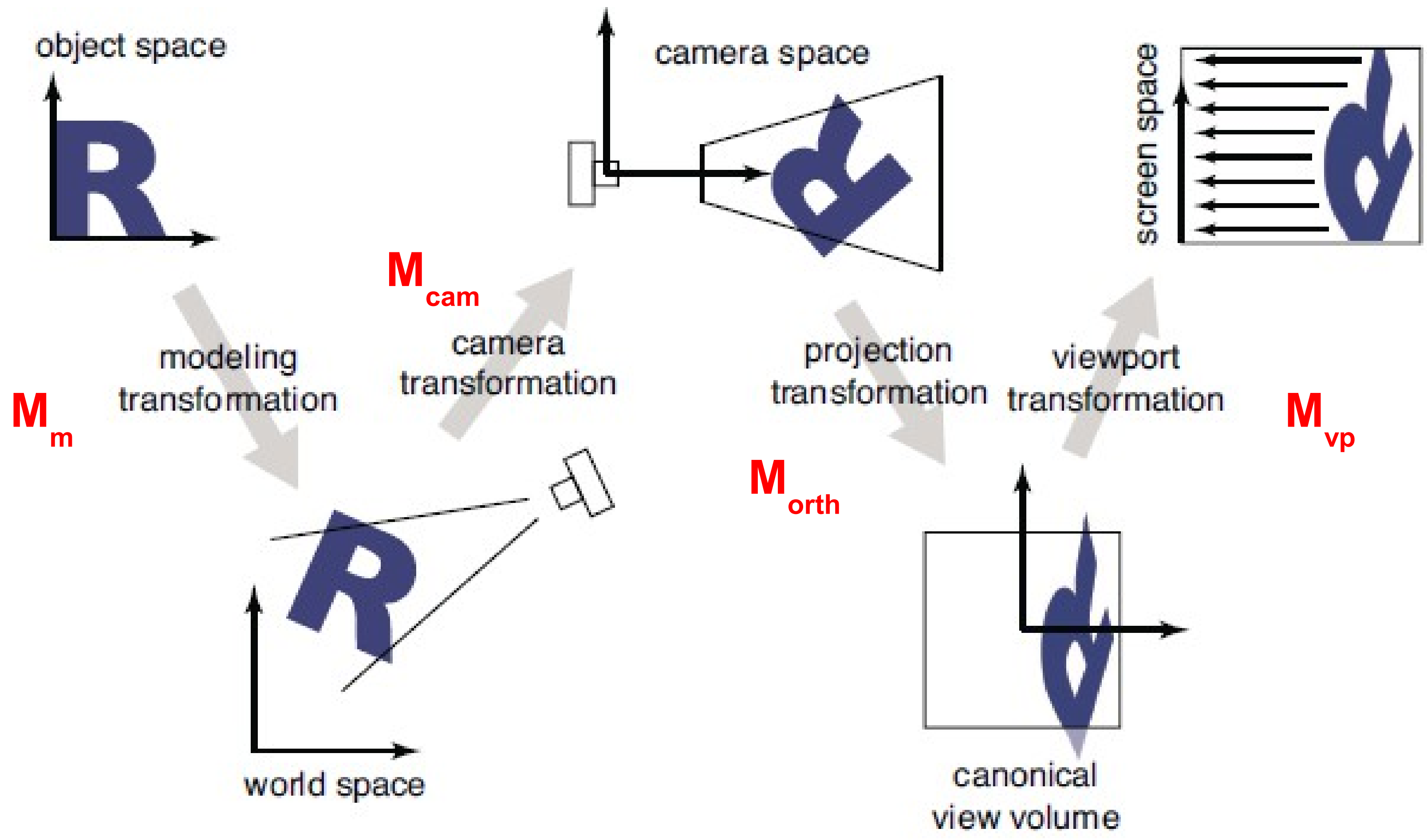


$$\begin{bmatrix} x_{cam} \\ y_{cam} \\ z_{cam} \\ 1 \end{bmatrix} = \begin{bmatrix} x_u & y_u & z_u & 0 \\ x_v & y_v & z_v & 0 \\ x_w & y_w & z_w & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & -x_e \\ 0 & 1 & 0 & -y_e \\ 0 & 0 & 1 & -z_e \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_w \\ y_w \\ z_w \\ 1 \end{bmatrix}$$

Camera Transformation



Modeling Transformation

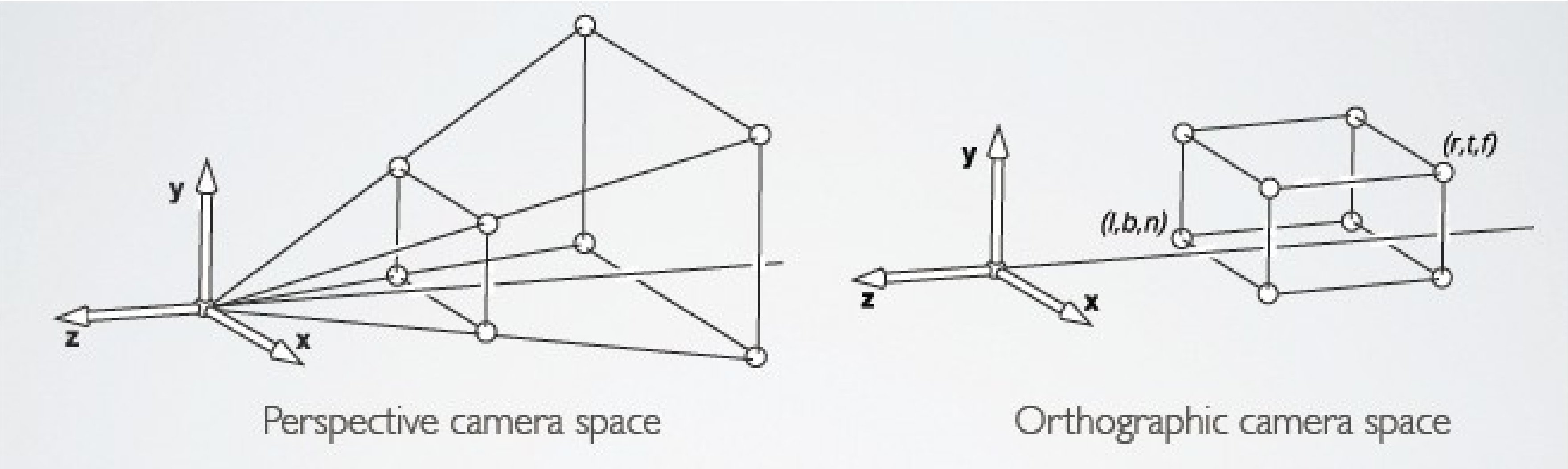


Orthographic Transformation

- Start with point in Object coordinates
- Convert to World Coordinates: M_m
- Convert to Camera Coordinates: M_{cam}
- Perform Orthographic Projection: M_{orth}
- Convert to Screen Coordinates: M_{vp}

$$\begin{bmatrix} x_s \\ y_s \\ z_c \\ 1 \end{bmatrix} = M_{vp} M_{orth} M_{cam} M_m \begin{bmatrix} x_o \\ y_o \\ z_o \\ 1 \end{bmatrix}$$

Perspective Projection

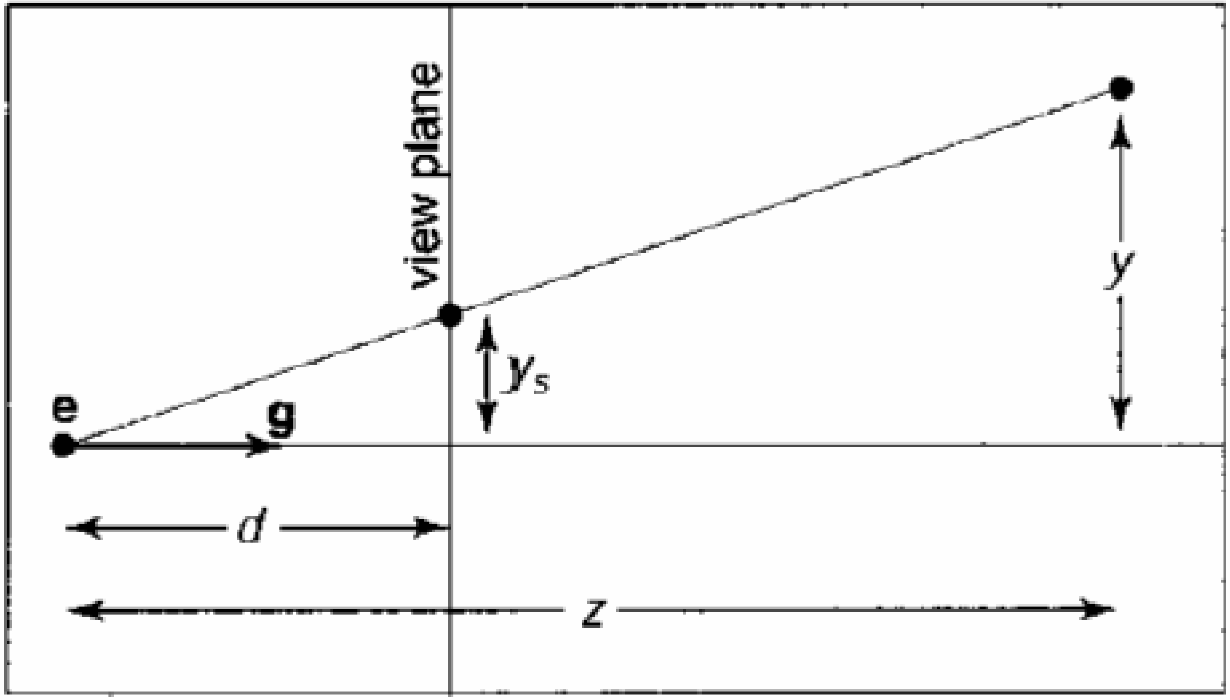


Perspective View Volume: Frustum　　　Orthographic View Volume

Projection lines go through the camera center !

Want to map the perspective view frustum onto the orthographic view volume

Perspective Projection



Similar Triangles

$$\frac{y_s}{d} = \frac{y}{z}$$

$$y_s = \frac{dy}{z}$$

How do we perform division?

Perspective Projection

Homogeneous Coordinates

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} \rightarrow \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

$$\begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \sim \begin{bmatrix} wx \\ wy \\ wz \\ w \end{bmatrix}$$

Allow any w

$$\begin{bmatrix} x \\ y \\ z \\ w \end{bmatrix} \sim \begin{bmatrix} x/w \\ y/w \\ z/w \\ 1 \end{bmatrix}$$

Divide by w to go back

What if w is zero?

Perspective Projection

$$y_s = \frac{dy}{z} \;\&\; x_s = \frac{dx}{z}$$

$$\begin{bmatrix} x_s \\ y_s \\ 1 \end{bmatrix} = \begin{bmatrix} dx/z \\ dy/z \\ 1 \end{bmatrix} \sim \begin{bmatrix} dx \\ dy \\ z \end{bmatrix} = \begin{bmatrix} d & 0 & 0 & 0 \\ 0 & d & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

What about z_s ?

Perspective Projection

$$\begin{bmatrix} x_s \\ y_s \\ z_s \\ 1 \end{bmatrix} \sim \begin{bmatrix} \tilde{x} \\ \tilde{y} \\ \tilde{z} \\ z \end{bmatrix} = \begin{bmatrix} d & 0 & 0 & 0 \\ 0 & d & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

$$z_s = 1$$

Z coordinate is lost ! How do we preserve it?

$$\begin{bmatrix} x_s \\ y_s \\ z_s \\ 1 \end{bmatrix} \sim \begin{bmatrix} \tilde{x} \\ \tilde{y} \\ \tilde{z} \\ z \end{bmatrix} = \begin{bmatrix} d & 0 & 0 & 0 \\ 0 & d & 0 & 0 \\ 0 & 0 & a & b \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

Perspective Projection

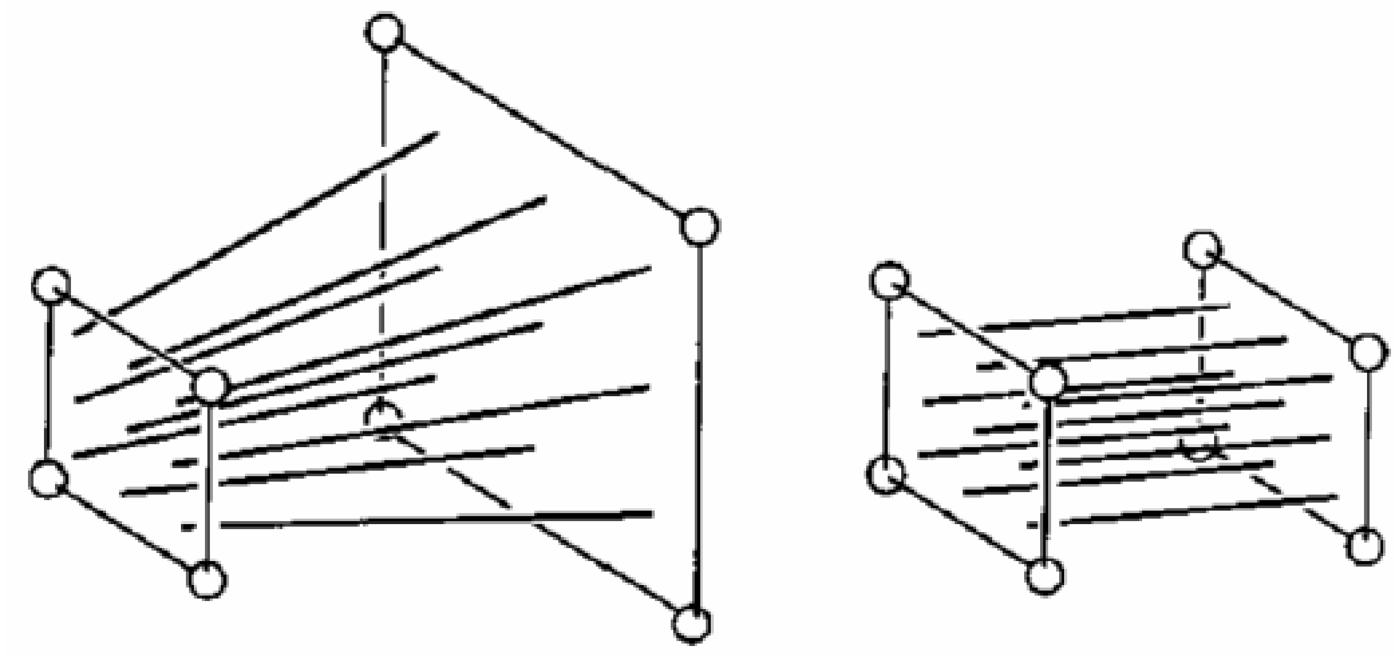
$$\begin{bmatrix} x_s \\ y_s \\ z_s \\ 1 \end{bmatrix} \sim \begin{bmatrix} \tilde{x} \\ \tilde{y} \\ \tilde{z} \\ z \end{bmatrix} = \begin{bmatrix} d & 0 & 0 & 0 \\ 0 & d & 0 & 0 \\ 0 & 0 & a & b \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

$$\tilde{z} = az + b \;\; \& \;\; z_s = \frac{az+b}{z}$$

Set $d = n$ and find a & b such that:
- when $z = n$ we get $z_s = n$
- when $z = f$ we get $z_s = f$

$$a = n + f \;\; \& \;\; b = -fn$$

Perspective Projection



$$P = \begin{bmatrix} n & 0 & 0 & 0 \\ 0 & n & 0 & 0 \\ 0 & 0 & n+f & -fn \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

Maps lines through the origin to lines parallel to z-axis preserving the point at z=n

Perspective Transformation

- Start with point in Object coordinates
- Convert to World Coordinates: M_m
- Convert to Camera Coordinates: M_{cam}
- Perform Perspective Projection: P
- Perform Orthographic Projection: M_{orth}
- Convert to Screen Coordinates: M_{vp}

$$\begin{bmatrix} x_s \\ y_s \\ z_c \\ 1 \end{bmatrix} = M_{vp} M_{orth} P M_{cam} M_m \begin{bmatrix} x_o \\ y_o \\ z_o \\ 1 \end{bmatrix}$$

Drawing Lines

```
Compute M=M_vp M_orth P M_cam M_m
For each line segment (a, b)
  p = Ma
  q = Mb
  drawline(xp/hp, yp/hp, xq/hq, yq/hq)
```

$M = M_{vp} M_{orth} P M_{cam} M_m$

Recap

- Viewing
- Projections
  - Orthographic
  - Perspective
- Transformations Pipeline