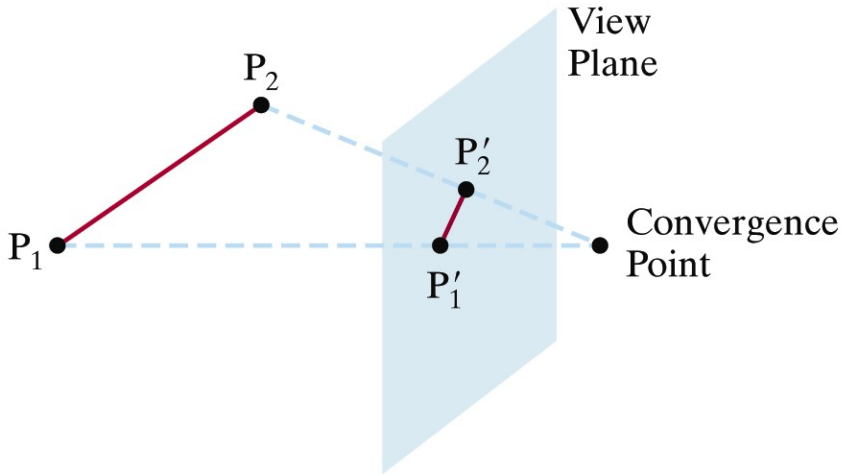
Computer Graphics 8:
Perspective Projections

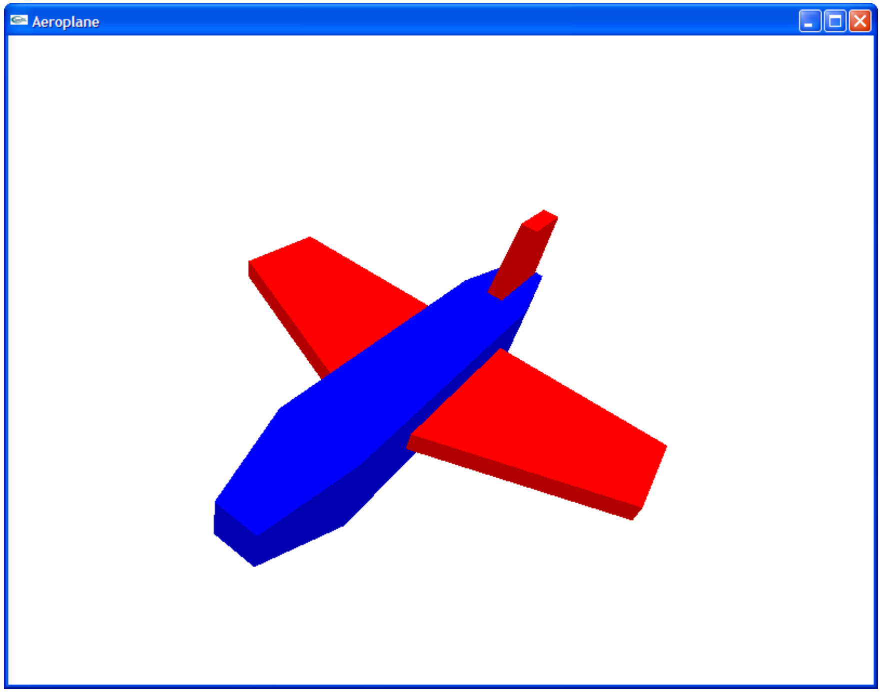
Contents

In today's lecture we are going to have a look at how perspective projections work in computer graphics

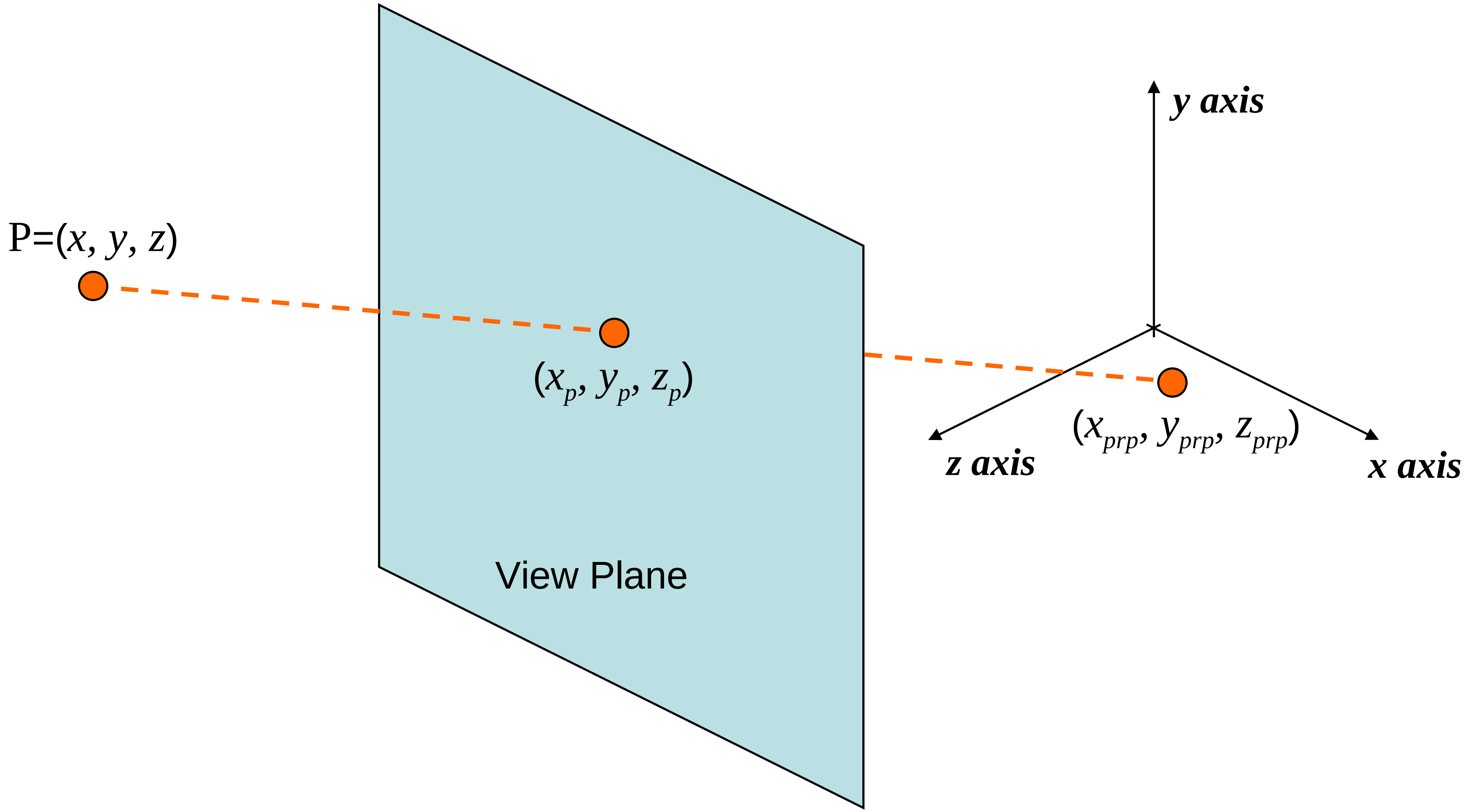
Perspective Projections

Remember the whole point of perspective projections





Projection Calculations



Projection Calculations (cont…)

Any point along the projector (x', y', z') can be given as:

$$x' = x - (x - x_{prp})u$$

$$y' = y - (y - y_{prp})u \qquad 0 \le u \le 1$$

$$z' = z - (z - z_{prp})u$$

When $u = 0$ we are at P, while when $u = 1$ we are at the *Projection Reference Point*

Projection Calculations (cont…)

At the view plane $z' = z_{vp}$ so we can solve the z' equation for u:

$$u = \frac{z_{vp} - z}{z_{prp} - z}$$

Projection Calculations (cont…)

Armed with this we can restate the equations for x' and y' for general perspective:

$$x_{vp} = x\left(\frac{z_{prp} - z_{vp}}{z_{prp} - z}\right) + x_{prp}\left(\frac{z_{vp} - z}{z_{prp} - z}\right)$$

$$y_{vp} = y\left(\frac{z_{prp} - z_{vp}}{z_{prp} - z}\right) + y_{prp}\left(\frac{z_{vp} - z}{z_{prp} - z}\right)$$

Perspective Projection Transformation Matrix

Because the x and y coordinates of a projected point are expressed in terms of z we need to do a little work to generate a perspective transformation matrix

First we use a homogeneous representation to give x_{vp} and y_{vp} as:

$$x_{vp} = \frac{x_h}{h} \qquad y_{vp} = \frac{y_h}{h}$$

where:

$$h = z_{prp} - z$$

Perspective Projection Transformation Matrix (cont…)

From the previous equations for x_{vp} and y_{vp} we can see that:

$$x_h = x(z_{prp} - z_{vp}) + x_{prp}(z_{vp} - z)$$
$$y_h = y(z_{prp} - z_{vp}) + y_{prp}(z_{vp} - z)$$

Perspective Projection Transformation Matrix (cont…)

Now we can set up a transformation matrix, that only contains perspective parameters, to convert a spatial position to homogeneous coordinates

First we calculate the homogeneous coordinates using the perspective-transformation matrix:

$$P_h = M_{pers} \cdot P$$

where P_h is the homogeneous point (x_h, y_h, z_h, h) and P is the coordinate position (x, y, z, 1)

Perspective Projection Transformation Matrix (cont…)

Setting up the matrix so that we calculate x_h and y_h is straightforward

However, we also need to preserve the z values – depth information

Otherwise the z coordinates are distorted by the homogeneous parameter h

We don't need to worry about the details here, but it means extra parameters (s_z and t_z) are added to the matrix

Perspective Projection Transformation Matrix (cont…)

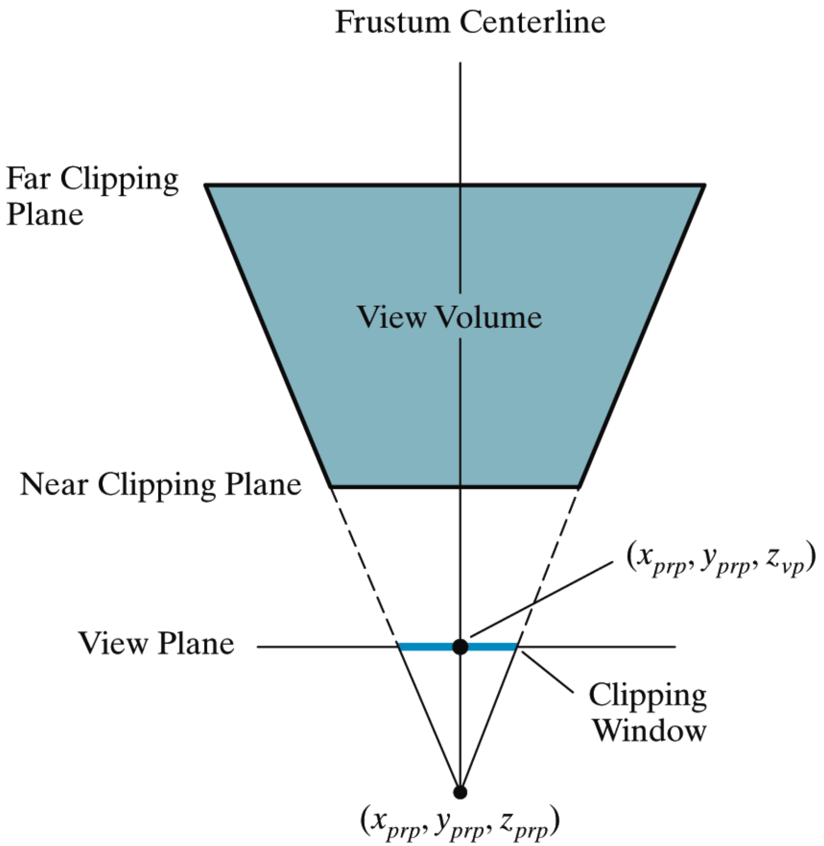
The following is the perspective projection matrix which arises:

$$M_{pers} = \begin{bmatrix} z_{prp} - z_{vp} & 0 & -x_{prp} & x_{prp} z_{prp} \\ 0 & z_{prp} - z_{vp} & -y_{prp} & y_{prp} z_{prp} \\ 0 & 0 & s_z & t_z \\ 0 & 0 & -1 & z_{prp} \end{bmatrix}$$

Setting Up A Perspective Projection

A perspective projection can be set up by specifying the position and size of the view plane and the position of the projection reference point

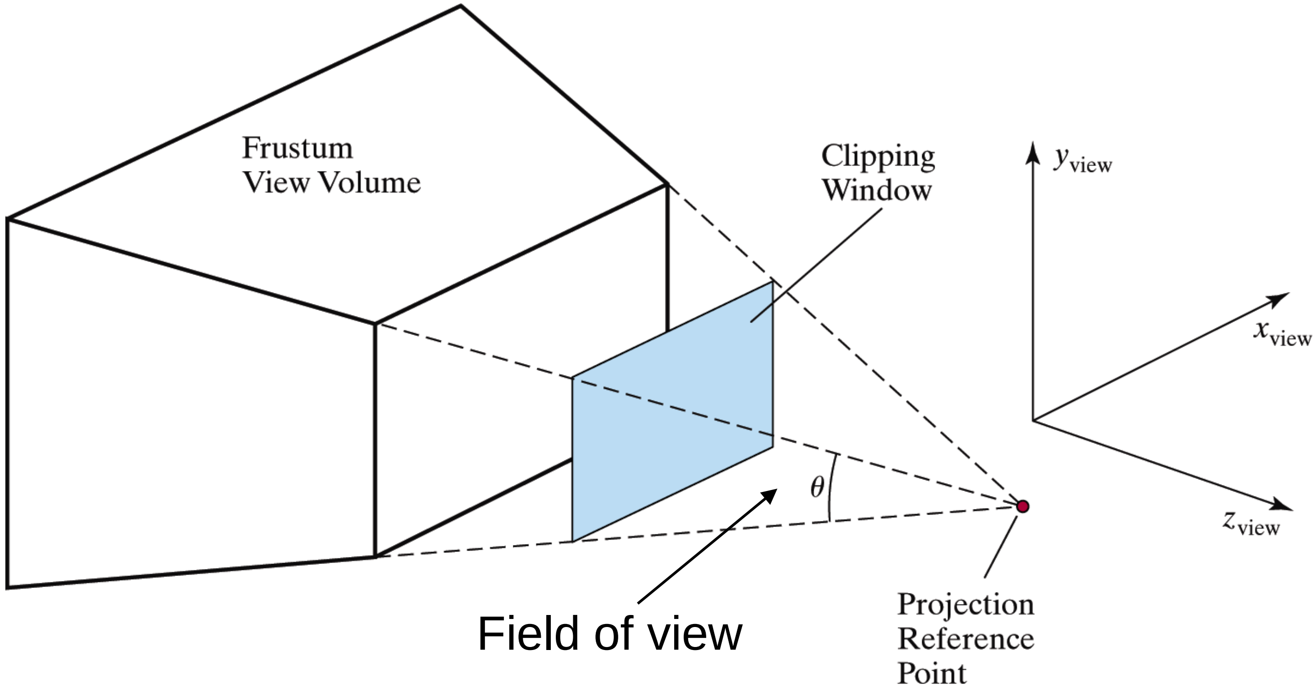
However, this can be kind of awkward



Setting Up A Perspective Projection (cont…)

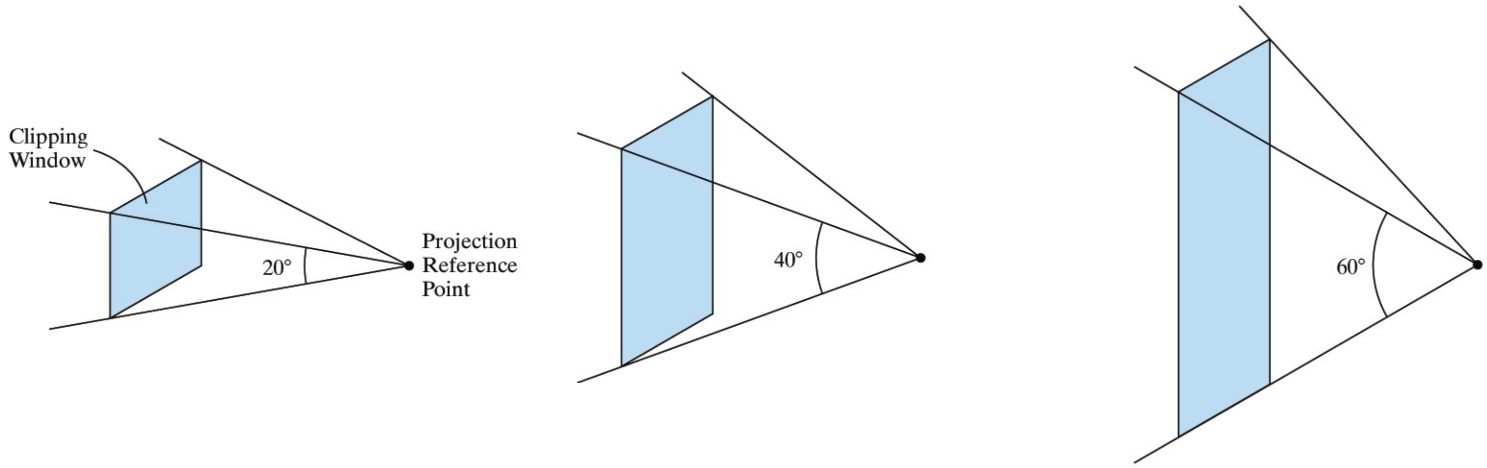
The *field of view* angle can be a more intuitive way to specify perspective projections

This is analogous to choosing a lense for a camera



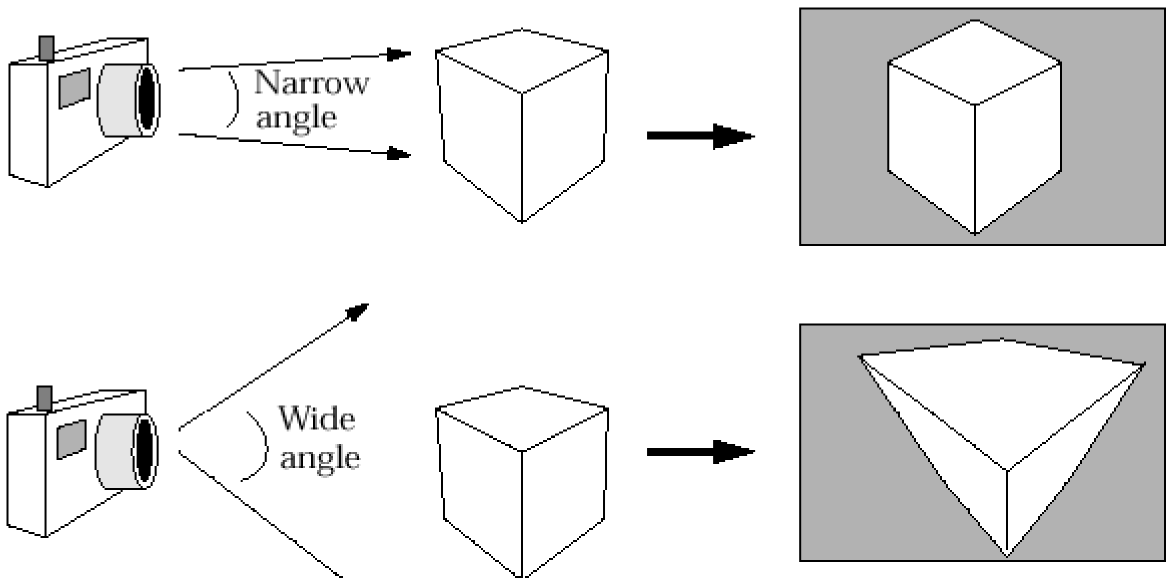
Setting Up A Perspective Projection (cont…)

Increasing the field of view angle increases the height of the view plane and so increases *foreshortening*



Setting Up A Perspective Projection (cont…)

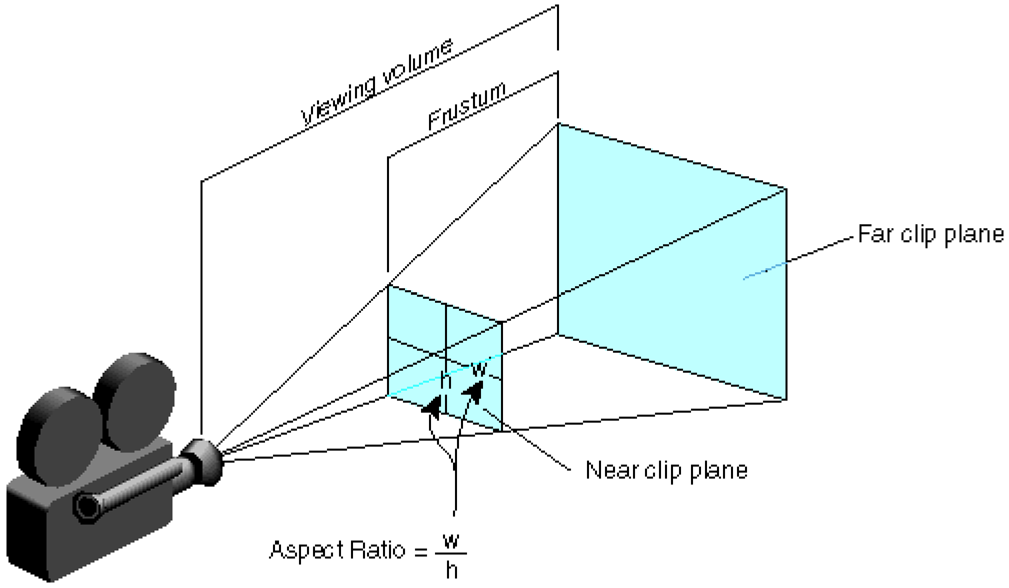
The amount of foreshortening that is present can greatly affect the appearance of our scenes



Setting Up A Perspective Projection (cont…)

We need one more thing to specify a perspective projections using the filed of view angle

The aspect ratio gives the ratio between the width sand height of the view plane



Summary

In today's class we looked at the detail of generating a perspective projection of a three dimensional scene