Computer Graphics

P. Healy

CS1-08
Computer Science Bldg.
tel: 202727
patrick.healy@ul.ie

Spring 2021–2022

Outline

Outline

Transforming Perspective Coordinates (contd.)

- We wish to find the transformation matrix to transform the frustrum to a **parallelepiped** (e.g., a brick)
- The problem with the equations is that they contain a with z in denominator
- We can get around this by **homogenising** the coordinates by multiplying by the $(z_P - z)$ term

$$x_h = x(z_P - z_{vp}) + x_P(z_{vp} - z)$$
$$y_h = y(z_P - z_{vp}) + y_P(z_{vp} - z)$$

where

$$x_p = \frac{x_h}{h}, \quad y_p = \frac{y_h}{h}, \quad h = z_P - z$$

- That is, given a point p we convert it with

$$p_h = M_p \cdot p$$

where $p_h = (x_h, y_h, z_h, h)$ and $p = (x, y, z, 1)$

Transforming Perspective Coordinates (contd.)

- The matrix M_p can then be concatenated in to the pipeline of other transformation matrices and the composite matrix is applied to to produce
- A side effect of this is that the z coordinates get distorted so scale and translation terms, s_z and t_z need to be introduced to compensate; they get determined at the normalization step later. So

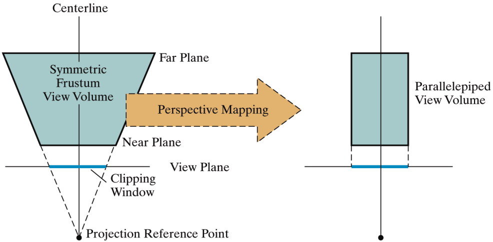
$$M_p = \begin{pmatrix} z_P - z_{vp} & 0 & -x_P & -x_P z_P \\ 0 & z_P - z_{vp} & -y_P & -y_P z_P \\ 0 & 0 & s_z & t_z \\ 0 & 0 & -1 & z_P \end{pmatrix}$$

Transforming Perspective Coordinates (contd.)

- M_p on previous slide converts a scene into homogeneous **parallel-projection** coordinates (see also p. 239, §5-2)
- When all is right in the world the perspective mapping derived from M_p gives the following:
- (In general we could have an **oblique** frustrum This gets transformed to an oblique (tilted) parallelepiped)

Transforming Perspective Coordinates (contd.)

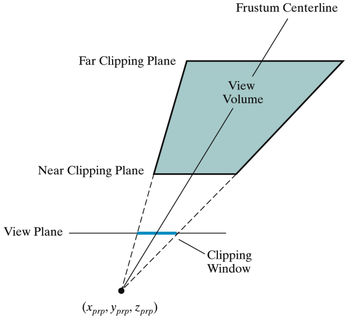
- M_p on previous slide converts a scene into homogeneous **parallel-projection** coordinates (see also p. 239, §5-2)
- When all is right in the world the perspective mapping derived from M_p gives the following:



- (In general we could have an **oblique** frustrum This gets transformed to an oblique (tilted) parallelepiped)

Transforming Perspective Coordinates (contd.)

- M_p on previous slide converts a scene into homogeneous **parallel-projection** coordinates (see also p. 239, §5-2)
- When all is right in the world the perspective mapping derived from M_p gives the following:
- (In general we could have an **oblique** frustrum This gets transformed to an oblique (tilted) parallelepiped)

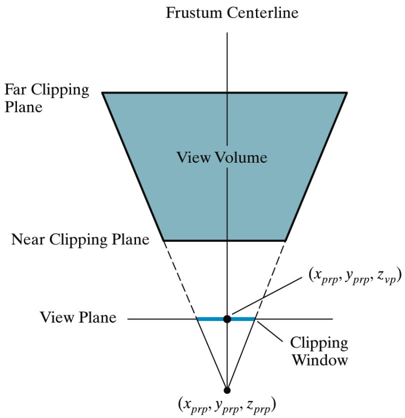


Symmetric Perspective-Projection Frustrum

- Due to positioning of clipping window an oblique frustrum may result
- The resulting oblique parallelepiped can be righted by applying a “reverse shear”
- So we need just worry about clipping windows that are aligned with the PRP
- For a given clipping window the BL, TR corner points are

$$(x_P - W/2, y_P - H/2)$$

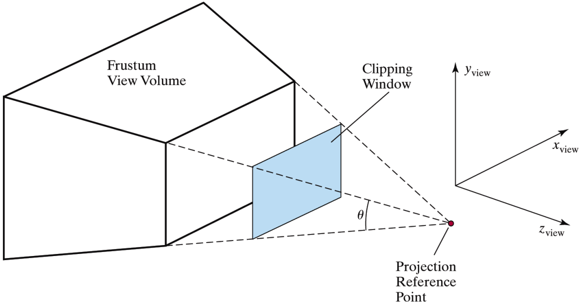
$$(x_P + W/2, y_P + H/2)$$



Note: in all of these slides I have used x_P whereas the book’s illustrations show x_{prp}; likewise y_P and z_P

Symmetric Perspective-Projection Frustrum (contd.)

- An alternative way to specify *symmetric* [   ] is by **field-of-view angle**



- When given [   ] and view-plane position, the height of the clipping window is determined

$$\tan\left(\frac{\theta}{2}\right) = \frac{height/2}{z_P - z_{vp}}$$

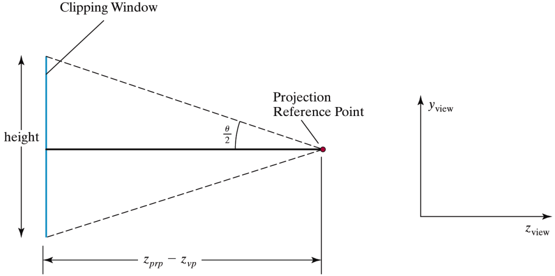
Symmetric Perspective-Projection Frustrum (contd.)

- An alternative way to specify *symmetric* is by **field-of-view angle**
- When given and view-plane position, the height of the clipping window is determined

$$\tan\left(\frac{\theta}{2}\right) = \frac{height/2}{z_P - z_{vp}}$$

Symmetric Perspective-Projection Frustrum (contd.)

- An alternative way to specify *symmetric* is by **field-of-view angle**
- When given and view-plane position, the height of the clipping window is determined



$$\tan\left(\frac{\theta}{2}\right) = \frac{height/2}{z_P - z_{vp}}$$

Symmetric Perspective-Projection Frustrum (contd.)

- In order to specify the clipping window we need to give from *width*, *height* or *aspect ratio (AR)*, since knowing two yields the third
- Rearranging the previous equation

$$\tan\left(\frac{\theta}{2}\right) = \frac{height/2}{z_P - z_{vp}}$$

we can get

$$height = 2(z_P - z_{vp})\tan\left(\frac{\theta}{2}\right)$$

and

$$\begin{aligned} z_P - z_{vp} &= \frac{height}{2}\cot\left(\frac{\theta}{2}\right) \\ &= \frac{width}{2AR}\cot\left(\frac{\theta}{2}\right) \end{aligned}$$

Symmetric Perspective-Projection Frustrum (contd.)

- Another treatment of the pipeline