

4.7.2 OpenGL Perspective Transformations

The function `Frustum` does not restrict the view volume to a symmetric (or right) frustum. The parameters are as shown in Figure 4.40. We can form the perspective matrix by first converting this frustum to the symmetric frustum with 45-degree sides (see Figure 4.39). The process is similar to the conversion of an oblique parallel view to an orthogonal view. First, we do a shear to convert the asymmetric frustum to a symmetric one. Figure 4.40 shows the desired transformation. The shear angle is determined by our desire to skew (shear) the point $((\text{left} + \text{right})/2, (\text{top} + \text{bottom})/2, -\text{near})$ to $(0, 0, -\text{near})$. The required shear matrix is

$$H(\theta, \phi) = H \left(\cot^{-1} \left(\frac{\text{left} + \text{right}}{-2\text{near}} \right), \cot^{-1} \left(\frac{\text{top} + \text{bottom}}{-2\text{near}} \right) \right)$$

The resulting frustum is described by the planes

$$x = \pm \frac{\text{right} - \text{left}}{-2 * \text{near}},$$

$$y = \pm \frac{\text{top} - \text{bottom}}{-2 * \text{near}},$$

$$z = -\text{near},$$

$$z = -\text{far}.$$

The next step is to scale the sides of this frustum to

$$x = \pm z,$$

$$y = \pm z,$$

without changing either the near plane or the far plane. The required scaling matrix is $S(-2 * \text{near}/(\text{right} - \text{left}), -2 * \text{near}/(\text{top} - \text{bottom}), 1)$. Note that this transformation is determined uniquely without reference to the location of the far plane $z = -\text{far}$ because in three dimensions, an affine transformation is determined by the results of the transformation on four points. In this case, these points are the four vertices where the sides of the frustum intersect the near plane.

To get the far plane to the plane $z = -1$ and the near plane to $z = 1$ after applying a projection normalization, we use the projection-normalization matrix **N**:

$$N = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \alpha & \beta \\ 0 & 0 & -1 & 0 \end{bmatrix},$$

with α and β as in Section 4.7.1. The resulting projection matrix is in terms of the near and far distances,

$$P = NSH = \begin{bmatrix} \frac{2 * \text{near}}{\text{right} - \text{left}} & 0 & \frac{\text{right} + \text{left}}{\text{right} - \text{left}} & 0 \\ 0 & \frac{2 * \text{near}}{\text{top} - \text{bottom}} & \frac{\text{top} + \text{bottom}}{\text{top} - \text{bottom}} & 0 \\ 0 & 0 & -\frac{\text{far} + \text{near}}{\text{far} - \text{near}} & \frac{-2\text{far} * \text{near}}{\text{far} - \text{near}} \\ 0 & 0 & -1 & 0 \end{bmatrix}$$