system superimposed on the face of a display (see Fig. 5.17), since a left-handed system gives the natural interpretation that larger z values are farther from the viewer. Notice that, in a left-handed system, positive rotations are *clockwise* when we are looking from a positive axis toward the origin. This definition of positive rotations allows the same rotation matrices given in this section to be used for either right- or left-handed coordinate systems. Conversion from right to left and from left to right is discussed in Section 5.9.

Translation in 3D is a simple extension from that in 2D:

$$T(d_z, d_y, d_z) = \begin{bmatrix} 1 & 0 & 0 & d_z \\ 0 & 1 & 0 & d_y \\ 0 & 0 & 1 & d_z \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{5.37}$$

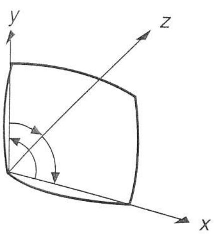

Figure 5.17
The left-handed coordinate system, with a superimposed display screen.

That is, $T(d_x, d_y, d_z) \cdot [x \;\; y \;\; z \;\; 1]^{\mathrm{T}} = [x + d_x \;\; y + d_y \;\; z + d_z \;\; 1]^{\mathrm{T}}$.

Scaling is similarly extended:

$$S(s_x, s_y, s_z) = \begin{bmatrix} s_z & 0 & 0 & 0 \\ 0 & s_y & 0 & 0 \\ 0 & 0 & s_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{5.38}$$

Checking, we see that $S(s_x, s_y, s_z) \cdot [x \;\; y \;\; z \;\; 1]^{\mathrm{T}} = [s_x \cdot x \;\; s_y \cdot y \;\; s_z \cdot z \;\; 1]^{\mathrm{T}}$.

The 2D rotation of Eq. (5.26) is just a 3D rotation about the z axis, which is

$$R_z(\theta) = \begin{bmatrix} \cos\theta & -\sin\theta & 0 & 0 \\ \sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{5.39}$$

This observation is easily verified: A 90° rotation of $[1 \;\; 0 \;\; 0 \;\; 1]^{\mathrm{T}}$, which is the unit vector along the x axis, should produce the unit vector $[0 \;\; 1 \;\; 0 \;\; 1]^{\mathrm{T}}$ along the y axis. Evaluating the product

$$\begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \end{bmatrix} \tag{5.40}$$

gives the predicted result of $[0 \;\; 1 \;\; 0 \;\; 1]^{\mathrm{T}}$.

The x-axis rotation matrix is

$$R_x(\theta) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta & 0 \\ 0 & \sin\theta & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{5.41}$$

The y-axis rotation matrix is

$$R_y(\theta) = \begin{bmatrix} \cos\theta & 0 & \sin\theta & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\theta & 0 & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{5.42}$$

The columns (and the rows) of the upper-left 3×3 submatrix of $R_z(\theta)$, $R_x(\theta)$, and $R_y(\theta)$ are mutually perpendicular unit vectors and the submatrix has a determinant of 1, which means the three matrices are special orthogonal, as discussed in Section 5.3. Also, the upper-left 3×3 submatrix formed by an arbitrary sequence of rotations is special orthogonal. Recall that orthogonal transformations preserve distances and angles.

All these transformation matrices have inverses. We obtain the inverse for T by negating d_x, d_y, and d_z; and that for S, by replacing s_x, s_y, and s_z by their reciprocals; we obtain the inverse for each of the three rotation matrices by negating the angle of rotation.

The inverse of any orthogonal matrix B is just B's transpose: $B^{-1} = B^{\mathrm{T}}$. In fact, taking the transpose does not need to involve even exchanging elements in the array that stores the matrix—it is necessary only to exchange row and column indexes when accessing the array. Notice that this method of finding an inverse is consistent with the result of negating θ to find the inverse of R_x, R_y, and R_z.

Any number of rotation, scaling, and translation matrices can be multiplied together. The result always has the form

$$M = \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{5.43}$$

As in the 2D case, the 3×3 upper-left submatrix R gives the aggregate rotation and scaling, whereas T gives the subsequent aggregate translation. We achieve some computational efficiency by performing the transformation explicitly as

$$\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = R \cdot \begin{bmatrix} x \\ y \\ z \end{bmatrix} + T, \tag{5.44}$$

where R and T are submatrices from Eq. (5.43).

Corresponding to the two-dimensional shear matrices in Section 5.2 are three 3D shear matrices. The (x, y) shear is

$$SH_{xy}(sh_x, sh_y) = \begin{bmatrix} 1 & 0 & sh_z & 0 \\ 0 & 1 & sh_y & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{5.45}$$

Applying SH_{xy} to the point $[x \;\; y \;\; z \;\; 1]^{\mathrm{T}}$, we have $[x + sh_x \cdot z \quad y + sh_y \cdot z \quad z \quad 1]^{\mathrm{T}}$. Shears along the x and y axes have a similar form.

So far, we have focused on transforming individual points. We transform lines, these being defined by two points, by transforming the endpoints. Planes, if they are defined by three points, may be handled the same way, but usually they are defined by a plane equation, and the coefficients of this plane equation must be transformed differently. We may also need to transform the plane normal. Let a plane be represented as the column vector of plane-equation coefficients $N = [A\ B\ C\ D]^T$. Then, a plane is defined by all points P such that $N \cdot P = 0$, where the symbol "$\cdot$" is the vector dot product and $P = [x\ y\ z\ 1]^T$. This dot product gives rise to the familiar plane equation $A\,x + B\,y + C\,z + D = 0$, which can also be expressed as the product of the row vector of plane-equation coefficients times the column vector P: $N^T \cdot P = 0$. Now suppose that we transform all points P on the plane by some matrix M. To maintain $N^T \cdot P = 0$ for the transformed points, we would like to transform N by some (to be determined) matrix Q, giving rise to the equation $(Q \cdot N)^T \cdot M \cdot P = 0$. This expression can in turn be rewritten as $N^T \cdot Q^T \cdot M \cdot P = 0$, using the identity $(Q \cdot N)^T = N^T \cdot Q^T$. The equation will hold if $Q^T \cdot M$ is a multiple of the identity matrix. If the multiplier is 1, this situation leads to $Q^T = M^{-1}$, or $Q = (M^{-1})^T$. Thus, the column vector N' of coefficients for a plane transformed by M is given by

$$N' = (M^{-1})^T \cdot N. \tag{5.46}$$

The matrix $(M^{-1})^T$ does not need to exist, in general, because the determinant of M might be zero. This situation would occur if M includes a projection (we might want to investigate the effect of a perspective projection on a plane).

If just the normal of the plane is to be transformed (e.g., to perform the shading calculations discussed in Chapter 14) and if M consists of only the composition of translation, rotation, and uniform scaling matrices, then the mathematics is even simpler. The N' of Eq. (5.46) can be simplified to $[A'\ B'\ C'\ 0]^T$. (With a zero W component, a homogeneous point represents a point at infinity, which can be thought of as a direction.)

5.8 COMPOSITION OF 3D TRANSFORMATIONS

In this section, we discuss how to compose 3D transformation matrices, using an example that will be useful in Section 6.5. The objective is to transform the directed line segments P_1P_2 and P_1P_3 in Fig. 5.18 from their starting position in part (a) to their ending position in part (b). Thus, point P_1 is to be translated to the origin, P_1P_2 is to lie on the positive z axis, and P_1P_3 is to lie in the positive y-axis half of the (y, z) plane. The lengths of the lines are to be unaffected by the transformation.

Two ways to achieve the desired transformation are presented. The first approach is to compose the primitive transformations T, R_x, R_y, and R_z. This approach, although somewhat tedious, is easy to illustrate, and understanding it will help us to build an understanding of transformations. The second approach, using the properties of special orthogonal matrices described in Section 5.7, is explained more briefly but is more abstract.

To work with the primitive transformations, we again break a difficult problem into simpler subproblems. In this case, the desired transformation can be done in four steps:

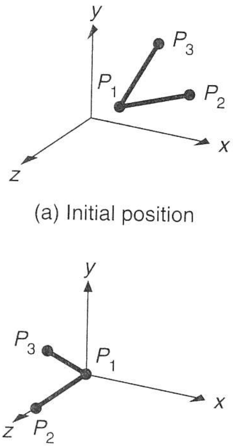

Figure 5.18
Transforming P_1, P_2, and P_3 from their initial position (a) to their final position (b).

1. Translate P_1 to the origin.
2. Rotate about the y axis such that P_1P_2 lies in the (y, z) plane.
3. Rotate about the x axis such that P_1P_2 lies on the z axis.
4. Rotate about the z axis such that P_1P_3 lies in the (y, z) plane.

Step 1: Translate P_1 to the origin. The translation is

$$T(-x_1, -y_1, -z_1) = \begin{bmatrix} 1 & 0 & 0 & -x_1 \\ 0 & 1 & 0 & -y_1 \\ 0 & 0 & 1 & -z_1 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{5.47}$$

Applying T to P_1, P_2, and P_3 gives

$$P_1' = T(-x_1, -y_1, -z_1) \cdot P_1 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}, \tag{5.48}$$

$$P_2' = T(-x_1, -y_1, -z_1) \cdot P_2 = \begin{bmatrix} x_2 - x_1 \\ y_2 - y_1 \\ z_2 - z_1 \\ 1 \end{bmatrix}, \tag{5.49}$$

$$P_3' = T(-x_1, -y_1, -z_1) \cdot P_3 = \begin{bmatrix} x_3 - x_1 \\ y_3 - y_1 \\ z_3 - z_1 \\ 1 \end{bmatrix}, \tag{5.50}$$

Step 2: Rotate about the y axis. Figure 5.19 shows P_1P_2 after step 1, along with the projection of P_1P_2 onto the (x, z) plane. The angle of rotation is $-(90 - \theta) = \theta - 90$. Then

$$\cos(\theta - 90) = \sin\theta = \frac{z_2'}{D_1} = \frac{z_2 - z_1}{D_1},$$

$$\sin(\theta - 90) = -\cos\theta = -\frac{x_2'}{D_1} = -\frac{x_2 - x_1}{D_1}, \tag{5.51}$$

where

$$D_1 = \sqrt{(z_2')^2 + (x_2')^2} = \sqrt{(z_2 - z_1)^2 + (x_2 - x_1)^2}\,. \tag{5.52}$$

When these values are substituted into Eq. (5.42), we get

$$P_2'' = R_y(\theta - 90) \cdot P_2' = [0 \;\; y_2 - y_1 \;\; D_1 \;\; 1]^{\mathrm{T}}. \tag{5.53}$$

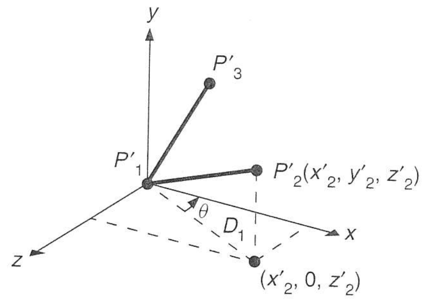

Figure 5.19 Rotation about the y axis: The projection of $P_1'P_2'$, which has length D_1, is rotated into the z axis. The angle θ shows the positive direction of rotation about the y axis: The actual angle used is $-(90-\theta)$.

As expected, the x component of P_2'' is zero, and the z component is the length D_1.

Step 3: Rotate about the x axis. Figure 5.20 shows P_1P_2 after step 2. The angle of rotation is ϕ, for which

$$\cos\phi = \frac{z_2''}{D_2'}, \quad \sin\phi = \frac{y_2''}{D_2'}, \tag{5.54}$$

where $D_2 = |P_1''P_2''|$, the length of the line $P_1''P_2''$. But the length of line $P_1''P_2''$ is the same as the length of line P_1P_2, because rotation and translation transformations preserve length, so

$$D_2 = |\ P_1''P_2''\ | = |\ P_1P_2\ | = \sqrt{(x_2-x_1)^2+(y_2-y_1)^2+(z_2-z_1)^2}\,. \tag{5.55}$$

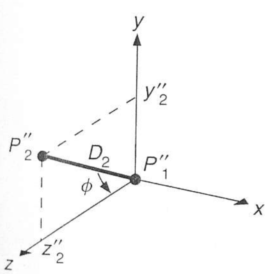

Figure 5.20
Rotation about the x axis: $P_1''P_2''$ is rotated into the z axis by the positive angle ϕ. D_2 is the length of the line segment. The line segment $P_1''P_3''$ is not shown, because it is not used to determine the angles of rotation. Both lines are rotated by $R_x(\phi)$.

The result of the rotation in step 3 is

$$\begin{aligned} P_2''' &= R_x(\phi)\cdot P_2'' = R_x(\phi)\cdot R_y(\theta-90)\cdot P_2' \\ &= R_x(\phi)\cdot R_y(\theta-90)\cdot T\cdot P_2 = [0 \quad 0 \quad |P_1P_2| \quad 1]^{\mathrm{T}}. \end{aligned} \tag{5.56}$$

That is, P_1P_2 now coincides with the positive z axis.

Step 4: Rotate about the z axis. Figure 5.21 shows P_1P_2 and P_1P_3 after step 3, with P_2''' on the z axis and P_3''' at the position

$$P_3''' = [x_3''' \quad y_3''' \quad z_3''' \quad 1]^{\mathrm{T}} = R_x(\phi)\cdot R_y(\theta-90)\cdot T(-x_1,-y_1,-z_1)\cdot P_3. \tag{5.57}$$

The rotation is through the positive angle α, with

$$\cos\alpha = y_3'''/D_3, \quad \sin\alpha = x_3'''/D_3, \quad D_3 = \sqrt{x_3'''^2 + y_3'''^2}\,. \tag{5.58}$$

Step 4 achieves the result shown in Fig. 5.18(b).

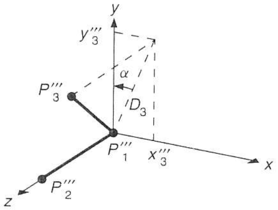

Figure 5.21
Rotation about the z axis: The projection of $P_1'P_3'$, whose length is D_3, is rotated by the positive angle α into the y axis, bringing the line itself into the (y, z) plane. D_3 is the length of the projection.

The composite matrix

$$R_z(\alpha) \cdot R_x(\phi) \cdot R_y(\theta - 90) \cdot T(-x_1, -y_1, -z_1) = R \cdot T \tag{5.59}$$

is the required transformation, with $R = R_z(\alpha) \cdot R_x(\phi) \cdot R_y(\theta - 90)$. We leave it to you to apply this transformation to P_1, P_2, and P_3, to verify that P_1 is transformed to the origin, P_2 is transformed to the positive z axis, and P_3 is transformed to the positive y half of the (y, z) plane.

The second way to obtain the matrix R is to use the properties of orthogonal matrices discussed in Section 5.3. Recall that the unit row vectors of R rotate into the principal axes. Replacing the second subscripts of Eq. (5.43) with x, y, and z for notational convenience

$$R = \begin{bmatrix} r_{1_x} & r_{2_x} & r_{3_x} \\ r_{1_y} & r_{2_y} & r_{3_y} \\ r_{1_z} & r_{2_z} & r_{3_z} \end{bmatrix}. \tag{5.60}$$

Because R_z is the unit vector along P_1P_2 that will rotate into the positive z axis,

$$R_z = [r_{1_z} \;\; r_{2_z} \;\; r_{3_z}]^{\mathrm{T}} = \frac{P_1P_2}{|\, P_1P_2 \,|}. \tag{5.61}$$

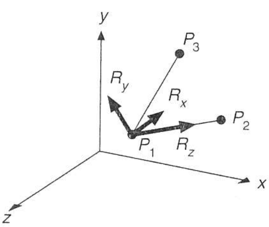

Figure 5.22
The unit vectors R_x, R_y, and R_z, which are transformed into the principal axes.

In addition, the R_x unit vector is perpendicular to the plane of P_1, P_2, and P_3 and will rotate into the positive x axis, so that R_x must be the normalized cross-product of two vectors in the plane:

$$R_x = [r_{1_x} \;\; r_{2_x} \;\; r_{3_x}]^{\mathrm{T}} = \frac{P_1P_3 \times P_1P_2}{|\, P_1P_3 \times P_1P_2 \,|}. \tag{5.62}$$

Finally,

$$R_y = [r_{1_y} \;\; r_{2_y} \;\; r_{3_y}]^{\mathrm{T}} = R_z \times R_x \tag{5.63}$$

will rotate into the positive y axis. The composite matrix is given by

$$\begin{bmatrix} r_{1x} & r_{2_x} & r_{3_x} & 0 \\ r_{1y} & r_{2_y} & r_{3_y} & 0 \\ r_{1z} & r_{2_z} & r_{3_z} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot T(-x_1, -y_1, -z_1) = R \cdot T, \tag{5.64}$$

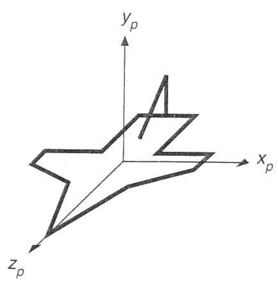

Figure 5.23
An airplane in the (x_p, y_p, z_p) coordinate system.

where R and T are as in Eq. (5.59). Figure 5.22 shows the individual vectors R_x, R_y, and R_z.

Now consider another example. Figure 5.23 shows an airplane defined in the x_p, y_p, z_p coordinate system and centered at the origin. We want to transform the airplane so that it heads in the direction given by the vector DOF (direction of flight), is centered at P, and is not banked, as shown in Fig. 5.24. The transformation to do this reorientation consists of a rotation to head the airplane in the proper direction, followed by a translation from the origin to P. To find the rotation matrix, we just determine in what direction each of the x_p, y_p, and z_p axes is

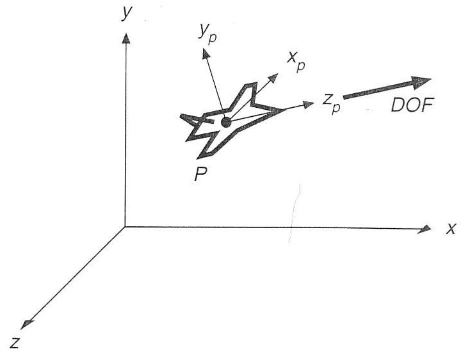

Figure 5.24 The airplane of Fig. 5.23 positioned at point *P*, and headed in direction *DOF*.

heading in Fig. 5.24, make sure the directions are normalized, and use these directions as column vectors in a rotation matrix.

The z_p axis must be transformed to the *DOF* direction, and the x_p axis must be transformed into a horizontal vector perpendicular to *DOF*—that is, in the direction of $y \times DOF$, the cross-product of y and DOF. The y_p direction is given by $z_p \times x_p = DOF \times (y \times DOF)$, the cross-product of z_p and x_p; hence, the three columns of the rotation matrix are the normalized vectors $|y \times DOF|$, $|DOF \times (y \times DOF)|$, and $|DOF|$:

$$R = \begin{bmatrix} & & & 0 \\ |\, y \times DOF \,| & |\, DOF \times (y \times DOF) \,| & |\, DOF \,| & 0 \\ & & & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{5.65}$$

The situation if *DOF* is in the direction of the y axis is degenerate, because there is an infinite set of possible vectors for the horizontal vector. This degeneracy is reflected in the algebra, because the cross-products $y \times DOF$ and $DOF \times (y \times DOF)$ are zero. In this special case, R is not a rotation matrix.

5.9 TRANSFORMATIONS AS A CHANGE IN COORDINATE SYSTEM

We have been discussing transforming a set of points belonging to an object into another set of points, when both sets are in the same coordinate system. With this approach, the coordinate system stays unaltered and the object is transformed with respect to the origin of the coordinate system. An alternative but equivalent way of thinking about a transformation is as a change of coordinate systems. This view is useful when multiple objects, each defined in its own local coordinate system, are