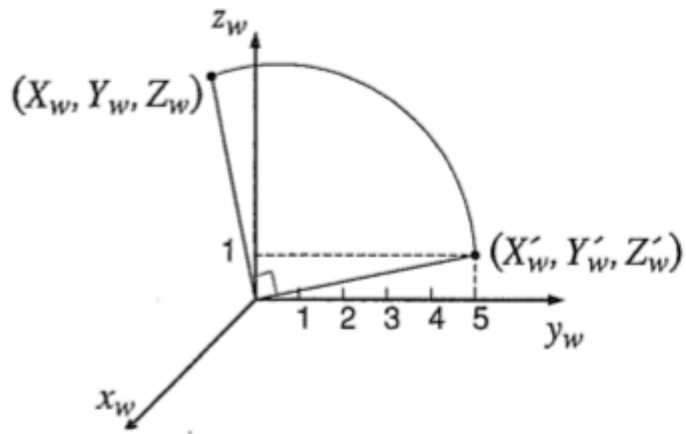
Transformations Examples

Example 1

An object in space is translated by 5 units in the y direction of the world coordinate system and then rotated by 90 degrees about the x axis of the world coordinate system. If a point on the object has the coordinates (0, 0, 1) with respect to its model coordinate system, what will be the world coordinates of the same point after the translation and the rotation?



ANSWER

The coordinates (X_w', Y_w', Z_w') after translation can be obtained by

$$\begin{aligned}[X_w' \ Y_w' \ Z_w' \ 1]^T &= Trans(0,5,0) \cdot [0 \ 0 \ 1 \ 1]^T \\ &= [0 \ 5 \ 1 \ 1]^T\end{aligned} \quad \text{(a)}$$

Then a rotation is applied:

$$[X_w \ Y_w \ Z_w \ 1]^T = Rot(x,90°) \cdot [0 \ 5 \ 1 \ 1]^T \quad \text{(b)}$$

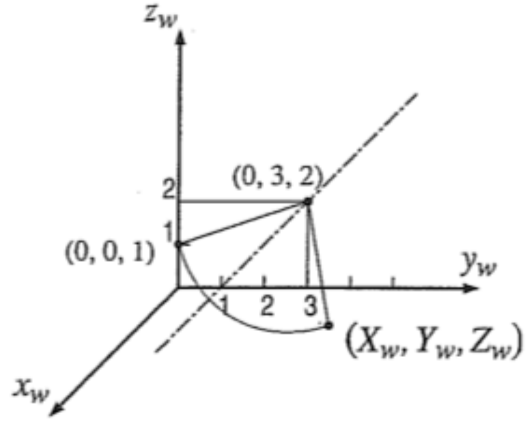
Thus the coordinates of the new point would be (0, –1, 5). Note that Equations (a) and (b) can be merged as follows:

$$[X_w \ Y_w \ Z_w \ 1]^T = Rot(x,90°) \cdot Trans(0,5,0) \cdot [0 \ 0 \ 1 \ 1]^T \quad \text{(c)}$$

Equation (c) is a much more convenient expression, especially when the coordinates of numerous points need to be calculated. In that case the transformation matrices $Rot(x,90°)$ and $Trans(0,5,0)$ are multiplied in advance to give an equivalent transformation matrix, and then the resulting matrix is applied to all the points involved. This process of calculating the equivalent transformation matrix by multiplying the associated transformation matrices in the proper sequence is called *concatenation*. This process is one of the benefits of using homogeneous coordinates, which enables the translation to be expressed by a matrix multiplication instead of an addition.

Example 2

An object in space is rotated by 90 degrees about an axis that is parallel to the x axis of the world coordinate system and passes through a point having world coordinates (0, 3, 2). If a point on the object has model coordinates (0, 0, 1), what will be the world coordinates of the same point after the rotation?



ANSWER

We have discussed rotations only about axes passing through the origin, so we have to move the object and the rotation axis together. The rotation axis must pass through the origin while the same relative position is maintained between the object and the rotation axis. Thus the object is translated by (0, –3, –2) together with the rotation axis so that the rotation axis coincides with the x axis of the world coordinate system. Then the object is rotated about the x axis by 90 degrees. Now the object is translated again, by (0, 3, 2), to return to the original position.

These operations can be expressed as

$$[X_w \;\; Y_w \;\; Z_w \;\; 1]^T = Trans(0,3,2) \cdot Rot(x,90°) \cdot Trans(0,-3,-2) \cdot [0\;0\;1\;1]^T \qquad \text{(d)}$$

Note the sequence of transformation matrices in Equation (d). The result can easily be verified by applying the transformations step by step, as in Example 3.1.

Expanding Equation (d) gives

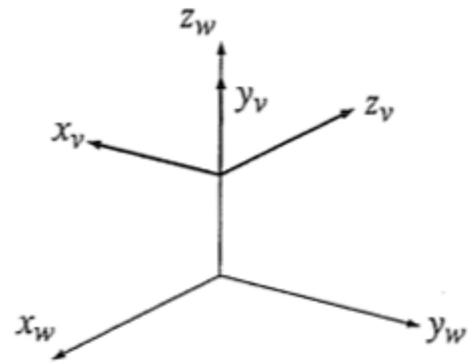
$$[X_w\, Y_w\, Z_w\, 1]^T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 3 \\ 0 & 0 & 1 & 2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos 90° & -\sin 90° & 0 \\ 0 & \sin 90° & \cos 90° & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & -3 \\ 0 & 0 & 1 & -2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \\ 1 \end{bmatrix}$$

$$= \begin{bmatrix} 0 & 4 & -1 & 1 \end{bmatrix}^T$$

This result reflects the operations illustrated in the accompanying figure.

Example 3

Corresponding to the viewpoint (–10, 0, 1), the viewsite (0, 0, 1), and the up vector (0, 0, 1), the viewing coordinate system is drawn as shown in the accompanying figure. Note that all the coordinate and component values are given in world coordinates. From the relative position between the viewing coordinate system and the world coordinate system, (i) calculate the mapping transformation T_{w-v} and (ii) calculate the coordinates of a point in viewing coordinates if it has world coordinates (5, 0, 1).



ANSWER

The first three numbers in the first column of T_{w-v} (i.e., n_x, n_y, and n_z), are (0 0 –1) because they are the x_v, y_v, and z_v components of the x_w axis. Similarly, o_x, o_y, and o_z, which are the x_v, y_v, and z_v components of the y_w axis, are (–1 0 0), and a_x, a_y, and a_z are (0 1 0). Because p_x, p_y, and p_z are, respectively, the x_v, y_v, and z_v coordinates of the origin of the $x_w y_w z_w$ coordinate system, and their values are 0, –1, and 0, respectively. Therefore we can derive T_{w-v} as follows:

$$T_{w-v} = \begin{bmatrix} 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & -1 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

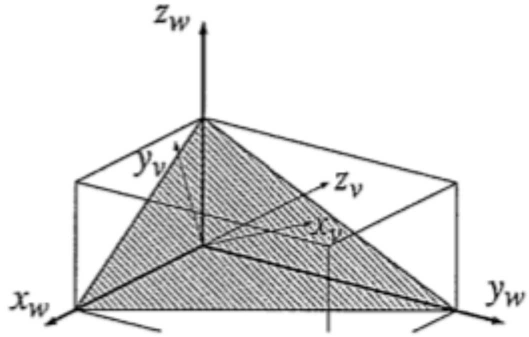
We obtain the viewing coordinates of (5, 0, 1) by applying T_{w-v} as follows:

$$\begin{bmatrix} X_v \\ Y_v \\ Z_v \\ 1 \end{bmatrix} = \begin{bmatrix} 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & -1 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 5 \\ 0 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ -5 \\ 1 \end{bmatrix}$$

Thus we can conclude that the viewing coordinates of (5, 0, 1) are (0, 0, –5), as can be guessed from the figure.

Example 4

The viewpoint and the viewsite are set at (5, 5, 5) and (0, 0, 0), respectively, to draw an isometric view, and the up vector is chosen to be (0, 0, 1). Derive the mapping transformation matrix T_{w-v}, and the viewing coordinates of a point represented by (0, 0, 5) in world coordinates.



ANSWER

The viewing coordinate system can be drawn as shown in the accompanying figure. The hatched triangle on which the *xv* and *yv* axes are placed indicates a plane parallel to the screen.

To derive the elements of T_{w-v}, we have to derive the x_v, y_v, and z_v components of each x_w axis, y_w axis, and z_w axis. For this purpose, we let each unit vector along the x_v, y_v, and z_v axes be $\mathbf{i_v}$, $\mathbf{j_v}$, and $\mathbf{k_v}$, respectively. Similarly the unit vectors along the x_w, y_w, and z_w axes are denoted **i**, **j**, and **k**, respectively.

The unit vector $\mathbf{k_v}$ acts in the direction from the viewsite to the viewpoint, so

$$\mathbf{k_v} = \frac{1}{\sqrt{3}}\mathbf{i} + \frac{1}{\sqrt{3}}\mathbf{j} + \frac{1}{\sqrt{3}}\mathbf{k}$$

As mentioned earlier in Section 3.2, the unit vector $\mathbf{j_v}$ should be in the direction of the projection of the up vector onto the screen. In other words, it will have the same direction as the vector obtained by subtracting, from the up vector, its component in the normal direction of the screen. Thus $\mathbf{j_v}$ can be expressed as follows if the up vector is denoted $\mathbf{u_p}$:

$$\mathbf{j_v} = \frac{\mathbf{u_p} - (\mathbf{u_p} \cdot \mathbf{k_v})\mathbf{k_v}}{\left|\mathbf{u_p} - (\mathbf{u_p} \cdot \mathbf{k_v})\mathbf{k_v}\right|} = \frac{-\frac{1}{3}\mathbf{i} - \frac{1}{3}\mathbf{j} + \frac{2}{3}\mathbf{k}}{\left|-\frac{1}{3}\mathbf{i} - \frac{1}{3}\mathbf{j} + \frac{2}{3}\mathbf{k}\right|} = -\frac{1}{\sqrt{6}}\mathbf{i} - \frac{1}{\sqrt{6}}\mathbf{j} + \frac{2}{\sqrt{6}}\mathbf{k}$$

The remaining unit vector $\mathbf{i}_v$ is obtained by the cross product:

$$\mathbf{i}_v = \mathbf{j}_v \times \mathbf{k}_v = \frac{1}{-\sqrt{2}}\mathbf{i} + \frac{1}{\sqrt{2}}\mathbf{j}$$

Now n_x, the x_v component of x_w axis, is derived as

$$\frac{1}{-\sqrt{2}} \text{ by } \mathbf{i} \cdot \mathbf{i}_v.$$

Similarly, n_y is

$$\frac{1}{-\sqrt{6}} \text{ by } \mathbf{i} \cdot \mathbf{j}_v.$$

and n_z is

$$\frac{1}{\sqrt{3}} \text{ by } \mathbf{i} \cdot \mathbf{k}_v.$$

The second and third columns of T_{w-v} are derived in the same way. We can ignore p_x, p_y, and p_z because the viewing coordinate system and the world coordinate system have the same origin in this example. Therefore, T_{w-v} is

$$T_{w-v} = \begin{bmatrix} -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0 & 0 \\ -\frac{1}{\sqrt{6}} & -\frac{1}{\sqrt{6}} & \frac{2}{\sqrt{6}} & 0 \\ \frac{1}{\sqrt{3}} & \frac{1}{\sqrt{3}} & \frac{1}{\sqrt{3}} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

and the viewing coordinates of (0, 0, 5) are

$$[X_v \;\; Y_v \;\; Z_v \;\; 1]^T = T_{w-v} \cdot [0 \;\; 0 \;\; 5 \;\; 1]^T = \left[0 \;\; \frac{5\sqrt{6}}{3} \;\; \frac{5\sqrt{3}}{3} \;\; 1\right]^T$$

The screen coordinates for the isometric view from the viewing coordinates are

$$\left(0 \;\; \frac{5\sqrt{6}}{3}\right)$$

simply by ignoring the z coordinates. The isometric projection is one of parallel projection, so any point on the z_w axis is projected onto the y axis of the screen coordinates. In fact, specifying the up vector to be (0, 0, 1) means that the z_w axis appears as a vertical line on the screen after projection.

Solution by independent axis rotation:

This method is based on the the principle that the set of transformations necessary to express the point in space as function of the VCS is the same used to transform the VCS so that it is coincident to the WCS :

```
theta = π / 4; (*first rotation angle around the z axis CCW, align the Zv to the Y-Z plane in radiants*)

Rz1 = {{Cos[theta], -Sin[theta], 0, 0}, {Sin[theta], Cos[theta], 0, 0}, {0, 0, 1, 0}, {0, 0, 0, 1}};
MatrixForm[Rz1] (* represent it in matrix form *)

alpha = 54.74 * π / 180; (* rotation around the X axis CCW, Zv coincident to Z, Xv=-X and Yv=-Y, in radiants *)
Rx = {{1, 0, 0, 0}, {0, Cos[alpha], -Sin[alpha], 0}, {0, Sin[alpha], Cos[alpha], 0}, {0, 0, 0, 1}};
MatrixForm[Rx]

phi = π; (*rotation again around Z, to bring VCS coincident to WCS, in radiants, note that that this
  is equivalent to mirroring in respect of X-Z and Y-Z planes, observe the matrix *)
Rz2 = {{Cos[phi], -Sin[phi], 0, 0}, {Sin[phi], Cos[phi], 0, 0}, {0, 0, 1, 0}, {0, 0, 0, 1}};
MatrixForm[Rz2]

K = Rz2.Rx.Rz1; (*this is our transformation matrix as product of the three rotations*)
MatrixForm[K]

P1 = {{0}, {0}, {5}, {1}}; (* point coordinate from WCS*)
P2 = K.P1 (* represent point from VCS*)
```

Out[27]//MatrixForm=

$$\begin{pmatrix} \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & 0 & 0 \\ \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

First rotation

Out[29]//MatrixForm=

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0.577288 & -0.816541 & 0 \\ 0 & 0.816541 & 0.577288 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Second rotation

Out[31]//MatrixForm=

$$\begin{pmatrix} -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Third rotation, or double mirroring

Out[32]//MatrixForm=

$$\begin{pmatrix} -0.707107 & 0.707107 & 0. & 0. \\ -0.408204 & -0.408204 & 0.816541 & 0. \\ 0.577382 & 0.577382 & 0.577288 & 0. \\ 0. & 0. & 0. & 1. \end{pmatrix}$$

Complete transformation matrix

Out[33]= {{0.}, {4.0827}, {2.88644}, {1.}}

Point coordinates from VCS