4
Transformations

The 1960s were the golden age of computer graphics. This was the time when many of its basic methods, algorithms, and techniques were developed, tested, and improved. Two of the most important concepts that were identified and studied in those years were transformations and projections. Workers in the graphics field immediately recognized the importance of transformations. Once a graphics object is constructed, the use of transformations enables the designer to create copies of the object and modify them in important ways. The necessity of projections was also realized early. Sophisticated graphics requires three-dimensional objects, but graphics output devices are two-dimensional. A three-dimensional object has to be projected on the flat output device in a way that will preserve its depth information. Thus, early researchers in computer graphics developed the mathematics of parallel and perspective projections and implemented these techniques. Nonlinear projections deform the projected image in various ways and are mostly used for artistic and ornamental purposes. These projections were also studied and implemented over the years by many people.

◇ **Exercise 4.1:** Most nonlinear projections are valued for their artistic and ornamental effects, but there is at least one type of nonlinear projection that has important practical applications. What is it?

The English term sea-change (or seachange) was coined by Shakespeare in his play *The Tempest*. The term means a gradual transformation in which the form is retained but the substance is replaced. Thus, sea-change is a real-life transformation. In computer graphics (and in other fields of science) the term transformation refers to a process that varies the location and orientation (i.e., the form) of an object while normally retaining its shape (i.e., substance) or at least its topology.

Today, transformations and projections are important components of computer graphics and computer-aided design (CAD). Transformations save the designer work and time, while projections are necessary because three-dimensional output devices are still rare (but see Section 6.15 for autostereoscopic displays, a revolutionary technique for

D. Salomon, *The Computer Graphics Manual*, Texts in Computer Science,
DOI 10.1007/978-0-85729-886-7_4, © Springer-Verlag London Limited 2011

three-dimensional displays), hence this part of the book.

Figure 4.1 shows the power of even the simplest two-dimensional transformations. It illustrates, from left to right, the following transformations: rotation, reflection, deformation (shearing), and scaling (see also Figure 4.3). It is not difficult to imagine the power of combining these transformations, but it is more difficult to imagine and visualize the power and flexibility of three-dimensional transformations.

Figure 4.1: Elementary Two-Dimensional Transformations.

The basic two-dimensional transformations are translation, rotation, reflection, scaling, and shearing. They are simple, but it is their combinations that make them powerful. It comes as a surprise to realize that these transformations can be specified by means of a single 3×3 matrix where only six of the nine elements are used. The same five basic transformations also exist in three dimensions, but have more degrees of freedom and therefore require more parameters to fully specify them. The general transformation matrix in three dimensions is 4×4, where 13 of the 16 elements control the transformations and the remaining three are used to specify the orientation of the projection plane in the case of perspective projections.

◇ **Exercise 4.2:** What transformations are possible in one dimension?

In contrast with the five basic transformations, there are more than five types of projections. As Figure 4.2 illustrates, we distinguish between linear and nonlinear projections. The former class consists of parallel and perspective projections, while the latter class includes many different types. Each type of projection has variants. Thus, parallel projections are classified into orthographic, axonometric, and oblique, while perspective projections include one-, two-, and three-point projections.

Nonlinear projections are all different and employ different approaches and ideas. Linear projections, on the other hand, are all based on the following simple rule of projection.

Rule. A three-dimensional object is projected on a two-dimensional plane called the projection plane. The object must be fully located on one side of the plane, and we imagine a viewer or an observer located on the other side. On that side, we select a point termed the *center of projection*, and it is the location of this point that determines the class, parallel or perspective, of the linear projection. A three-dimensional point $\mathbf{P}$ on the object is projected to a two-dimensional point $\mathbf{P}^*$ on the projection plane by connecting $\mathbf{P}$ to the center of projection with a straight segment. Point $\mathbf{P}^*$ is placed at the intersection of this segment with the projection plane. When the center of projection

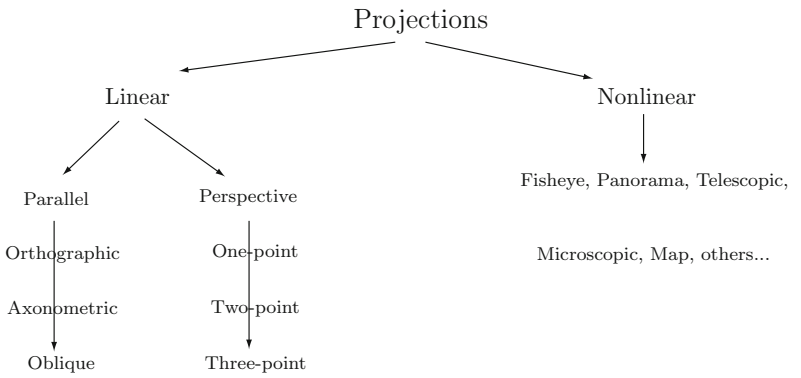


Figure 4.2: Classification of Projections.

is at infinity, the result is a parallel projection. If the center of projection is at the observer, the projection is perspective.

When working on computer graphics projects, we discover very quickly that transformations are an important part of the process of building an image. If an image has two identical (or even similar) parts, such as wheels, only one part need be constructed from scratch. The other parts can be obtained by copying the first and then moving, reflecting, and rotating it to bring it to the right shape, size, position, and orientation. Often, we want to zoom in on a small part of an image so that more detail can be seen. Sometimes it is useful to zoom out, so a large image can be seen in its entirety on the screen, even though no details can then be discerned. Operations such as moving, rotating, reflecting, or scaling an image are called *geometric transformations* and are discussed in this chapter for two and three dimensions.

4.1 Introduction

Mathematically, a geometric transformation is a function f whose domain and range are points. We denote by $\mathbf{P}$ a general point before any transformation and by $\mathbf{P}^*$ the same point after a transformation. The notation $\mathbf{P}^* = f(\mathbf{P})$ implies that the transformed point $\mathbf{P}^*$ is obtained by applying f to $\mathbf{P}$. We call our transformations *geometric* because they have geometric interpretations. Thus, only certain functions f can be used. Years of study and practical experience have shown that in order for it to be meaningful as a geometric transformation, a function must satisfy two conditions: it has to be *onto* and *one-to-one*.

- A general function f maps its domain D into its range R. If every point in R has a corresponding point in D, then the function maps its domain *onto* its range. An example is $f(x) = \lfloor x \rfloor$, which maps the real numbers onto the integers. Every integer has a real number (in fact, infinitely many real numbers) that map to it. Another example is $g(x) = 1/x$, a mapping from the real numbers into the real numbers. This mapping is

not onto because no real number maps to zero. Requiring a transformation to be onto makes sense since it guarantees that there will not be any special points $\mathbf{P}^*$ that cannot be reached by the transformation.

■ An arbitrary function may map two distinct points x and y into the same point. Function $f(x)$ above maps the two distinct numbers 9.2 and 9.9 into the integer 9. A *one-to-one* function satisfies $x \neq y \rightarrow f(x) \neq f(y)$. Function $g(x)$ above is one-to-one. The requirement that a transformation be one-to-one makes sense because it implies that a given point $\mathbf{P}^*$ is the transformed image of one point only, thereby making it possible to reconstruct the inverse transformation.

Definition. A geometric transformation is a function that is both onto and one-to-one, and whose range and domain are points.

◇ **Exercise 4.3:** Do either of the two real functions $f_1(x, y) = (x^2, y)$ and $f_2(x, y) = (x^3, y)$ satisfy the definition above?

There are two ways to look at geometric transformations. We can interpret them as either moving the points to new locations or as moving the entire coordinate system while leaving the points alone. The latter interpretation is discussed in Section 4.5, but the reader should realize that whatever interpretation is used, the movement caused by a geometric transformation is *instantaneous*. We should not think of a point as moving along a path from its original location to a new location, but rather as being grabbed and immediately planted in its new location.

> The description of right lines and circles, upon which geometry is founded, belongs to mechanics. Geometry does not teach us to draw these lines, but requires them to be drawn.
>
> —Isaac Newton (1687).

Combining transformations is an important operation that is discussed in detail in Section 4.2.2. This paragraph intends to make it clear that such a combination (sometimes called a *product*) amounts to a *composition* of functions. If functions f and g represent two transformations, then the composition $g \circ f$ represents the product of the two transformations. Such a composition is often written as $\mathbf{P}^* = g(f(\mathbf{P}))$. It can be shown that combining transformations is associative (i.e., $g \circ (f \circ h) = (g \circ f) \circ h$). This fact, together with a few other basic properties of transformations, makes it possible to identify *groups* of transformations. A discussion of mathematical groups is beyond the scope of this book but can be found in many texts on linear algebra. A set of transformations constitutes a group if it includes the identity transformation, if it is closed, and if every transformation in the set has an inverse that is also included in the set.

An example of a group of transformations is the set of two-dimensional rotations about the origin through angles of 0° and 180°. This two-element set is a group because a 0° rotation is an identity transformation and because a 180° rotation is the inverse of itself.

◇ **Exercise 4.4:** Is the operation of combining transformations commutative?

Another important example of a group of transformations is the set of *linear transformations* that map a point $\mathbf{P} = (x, y, z)$ to a point $\mathbf{P}^* = (x^*, y^*, z^*)$, where

$$\begin{aligned} x^* &= a_{11}x + a_{12}y + a_{13}z + a_{14}, \\ y^* &= a_{21}x + a_{22}y + a_{23}z + a_{24}, \\ z^* &= a_{31}x + a_{32}y + a_{33}z + a_{34}. \end{aligned} \tag{4.1}$$

Each new coordinate depends on all three original coordinates, and the dependence is linear. Such transformations are called *affine* and are defined more rigorously on Page 218.

A little thinking shows that the coefficients a_{i4} of Equation (4.1) represent quantities that are added to the transformed coordinates (x^*, y^*, z^*) regardless of the original coordinates, thereby simply *translating* $\mathbf{P}^*$ in space. This is why we start the detailed discussion here by temporarily ignoring these coefficients, which leads to the simple system of equations

$$\begin{aligned} x^* &= a_{11}x + a_{12}y + a_{13}z, \\ y^* &= a_{21}x + a_{22}y + a_{23}z, \\ z^* &= a_{31}x + a_{32}y + a_{33}z. \end{aligned} \tag{4.2}$$

If the 3×3 coefficient matrix of this system of equations is nonsingular or, equivalently, if the determinant of the coefficient matrix is nonzero (see any text on linear algebra for a refresher on matrices and determinants), then the system is easy to invert and can be expressed in the form

$$\begin{aligned} x &= b_{11}x^* + b_{12}y^* + b_{13}z^*, \\ y &= b_{21}x^* + b_{22}y^* + b_{23}z^*, \\ z &= b_{31}x^* + b_{32}y^* + b_{33}z^*, \end{aligned} \tag{4.3}$$

where the b_{ij}'s are expressed in terms of the a_{ij}'s. It is now easy to see that, for example, the two-dimensional line $Ax + By + C = 0$ is transformed by Equation (4.3) to the two-dimensional line

$$(Ab_{11} + Bb_{21})x^* + (Ab_{12} + Bb_{22})y^* + C = 0.$$

◇ **Exercise 4.5:** Show that Equation (4.3) maps the general second-degree curve

$$Ax^2 + Bxy + Cy^2 + Dx + Ey + F = 0$$

to another second-degree curve.

In general, an affine transformation maps any curve of degree n to another curve of the same degree.

4.2 Two-Dimensional Transformations

In practice, a complete two-dimensional image is constructed on the screen object-by-object and it may be edited before it is deemed satisfactory. One aspect of editing is to *transform* objects. Typical transformations (Figures 4.1 and 4.3 and color Plate Q.1) are moving or sliding (translation), reflecting or flipping (mirror image), zooming (scaling), rotating, and shearing (distorting). Notice how the orientation of Bach's nose in Figure 4.3 is different for reflection and rotation.

Figure 4.3: Two-Dimensional Transformations.

The transformation can be applied to every pixel of the object. Alternatively, it can be applied only to some key points that fully define the object (such as the four corners of a rectangle), following which the transformed object is constructed from the transformed key points.

> As soon as we use words like "image," we are already thinking of how one shape corresponds to the other—of how you might move one shape to bring it into coincidence with the other. Bilateral symmetry means that if you reflect the left half in a mirror, then you obtain the right half. Reflection is a mathematical concept, but it is not a shape, a number, or a formula. It is a *transformation*—that is, a rule for moving things around.
>
> —Ian Stewart, *Nature's Numbers* (1995).

The same principle applies to a three-dimensional image. Such an image consists of one or more three-dimensional objects that can be transformed individually, following which the entire image should be projected on the two-dimensional screen (or other output device). We first take a look at the mathematics of two-dimensional transformations.

We use the notation $\mathbf{P} = (x, y)$ for a point and $\mathbf{P}^* = (x^*, y^*)$ for the transformed point. We are looking for a simple, fast transformation rule, so it is natural to try a linear transformation (i.e., a mathematical rule that does not use operations more complex than multiplications and shifts). The simplest linear transformation is $x^* = ax + cy$ and $y^* = bx + dy$, in which each of the new coordinates is a linear combination of the two old coordinates. This transformation can be written $\mathbf{P}^* = \mathbf{PT}$, where $\mathbf{T}$ is the 2×2 matrix $\left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right)$. Thus, the transformation depends on just four parameters, which makes it easy to analyze and fully understand it.

To understand the effect of each of the four matrix elements, we start by setting $b = c = 0$. The transformation becomes $x^* = ax$ and $y^* = dy$, i.e., scaling. If applied to all the points of an object, all the x dimensions are scaled by a factor of a and all the y dimensions are scaled by a factor of d. Note that a and d can also be less than 1, which results in shrinking the object. If a or d (or both) equal -1, the transformation is a *reflection*. Any other negative values result in both scaling and reflection.

Note that scaling an object by factors of a and d changes its area by a factor of $a \times d$ and that this factor is also the value of the determinant of the scaling matrix $\left(\begin{smallmatrix} a & 0 \\ 0 & d \end{smallmatrix}\right)$.

Here are examples of scaling and reflection. In $\mathbf{A}$, the y coordinates are scaled by a factor of 2. In $\mathbf{B}$, the x coordinates are reflected. In $\mathbf{C}$, the x dimensions are shrunk to 0.001 of their original values. In $\mathbf{D}$, the figure is shrunk to a vertical line.

$$\mathbf{A} = \begin{pmatrix} 1 & 0 \\ 0 & 2 \end{pmatrix}, \quad \mathbf{B} = \begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix}, \quad \mathbf{C} = \begin{pmatrix} 0.001 & 0 \\ 0 & 1 \end{pmatrix}, \quad \mathbf{D} = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}.$$

◇ **Exercise 4.6:** In the novel *The Oxford Murders*, the author mentions the sequence of symbols **ΠΩ8**. Guess the meanings of the symbols and the next symbol in this sequence. (Hint. Ignore the obvious meanings of the `M` and the `8`. This has to do with symmetry, specifically, with reflection.)

◇ **Exercise 4.7:** What scaling transformation changes a circle to an ellipse?

The next step is to set $a = 1$ and $d = 1$ (no scaling or reflection) and explore the effect of matrix elements b and c. The transformation becomes $x^* = x + cy$, $y^* = bx + y$. We first set $b = 1$ and $c = 0$ and easily find out that matrix $\left(\begin{smallmatrix} 1 & 1 \\ 0 & 1 \end{smallmatrix}\right)$ transforms the four points $(1, 0)$, $(3, 0)$, $(1, 1)$, and $(3, 1)$ to $(1, 1)$, $(3, 3)$, $(1, 2)$, and $(3, 4)$, respectively. When we plot the original points and the transformed points (Figure 4.4a), it becomes obvious that the original rectangle has been sheared vertically and was transformed into a parallelogram. A similar shearing effect results from matrix $\left(\begin{smallmatrix} 1 & 0 \\ 1 & 1 \end{smallmatrix}\right)$. The quantities b and c are therefore responsible for shearing. Figure 4.4b shows the connection between shearing and the operation of scissors, which is the reason for the term shearing.

◇ **Exercise 4.8:** Apply the shearing transformation $\left(\begin{smallmatrix} 1 & -1 \\ 0 & 1 \end{smallmatrix}\right)$ to the four points $(1, 0)$, $(3, 0)$, $(1, 1)$, and $(3, 1)$. What are the transformed points? What geometrical figure do they represent?

The next important transformation is rotation. Figure 4.5 shows a point $\mathbf{P}$ rotated clockwise about the origin through an angle θ to become $\mathbf{P}^*$. Simple trigonometry yields

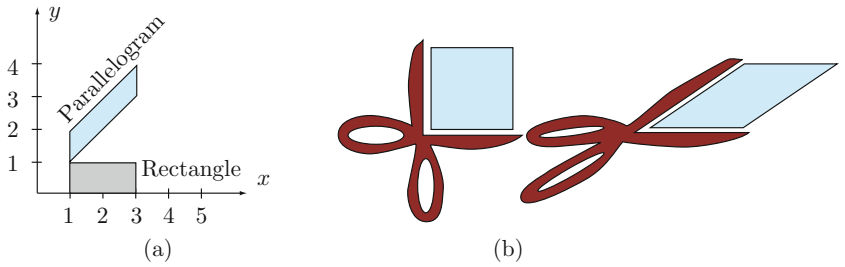


Figure 4.4: Scissors and Shearing.

$x = R\cos\alpha$ and $y = R\sin\alpha$. From this, we get the expressions for x^* and y^*

$$x^* = R\cos(\alpha - \theta) = R\cos\alpha\cos\theta + R\sin\alpha\sin\theta = x\cos\theta + y\sin\theta,$$
$$y^* = R\sin(\alpha - \theta) = -R\cos\alpha\sin\theta + R\sin\alpha\cos\theta = -x\sin\theta + y\cos\theta.$$

Hence, the clockwise rotation matrix in two dimensions is

$$\begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix}, \quad \begin{matrix} \text{which also} \\ \text{equals} \\ \text{the product} \end{matrix} \quad \begin{pmatrix} \cos\theta & 0 \\ 0 & \cos\theta \end{pmatrix} \begin{pmatrix} 1 & -\tan\theta \\ \tan\theta & 1 \end{pmatrix}. \tag{4.4}$$

This shows that any rotation in two dimensions is a combination of scaling (and, perhaps, reflection) by a factor of $\cos\theta$ and shearing, an unexpected result (that's true for all angles where $\tan\theta$ is finite).

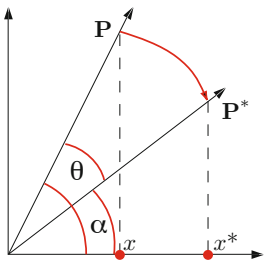


Figure 4.5: Clockwise Rotation.

◇ **Exercise 4.9:** Show how a 45° rotation can be achieved by scaling followed by shearing.

◇ **Exercise 4.10:** Discuss rotation in two dimensions using the polar coordinates (r, θ) of points instead of the Cartesian coordinates (x, y).

A rotation matrix has the following property: When any row is multiplied by itself, the result is 1, and when a row is multiplied by another row, the result is 0. The same is true for columns. Such a matrix is called *orthonormal*.

Matrix $\mathbf{T}_1$ below rotates counterclockwise. Matrix $\mathbf{T}_2$ reflects about the line $y = x$, and matrix $\mathbf{T}_3$ reflects about the line $y = -x$. Note the determinants of these matrices. In general, a determinant of $+1$ indicates pure rotation, whereas a determinant of -1 indicates pure reflection. (As a reminder, $\det \left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right) = ad - bc$.)

$$\mathbf{T}_1 = \begin{pmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{pmatrix}; \qquad \mathbf{T}_2 = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}; \qquad \mathbf{T}_3 = \begin{pmatrix} 0 & -1 \\ -1 & 0 \end{pmatrix}. \tag{4.5}$$

◇ **Exercise 4.11:** Show that a y-reflection (i.e., reflection about the x axis) followed by a reflection through the line $y = -x$ produces pure rotation.

◇ **Exercise 4.12:** Show that the transformation matrix

$$\begin{pmatrix} \frac{1-t^2}{1+t^2} & \frac{2t}{1+t^2} \\ \frac{-2t}{1+t^2} & \frac{1-t^2}{1+t^2} \end{pmatrix}$$

produces pure rotation.

◇ **Exercise 4.13:** For what values of A does the following matrix represent pure rotation and for what values does it represent pure reflection?

$$\begin{pmatrix} a/A & b/A \\ -b/A & a/A \end{pmatrix}.$$

▪ **A 90° Rotation:** For a 90° clockwise rotation, the rotation matrix is

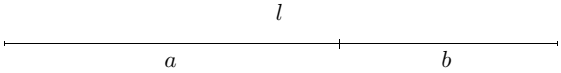
$$\begin{pmatrix} \cos(90) & -\sin(90) \\ \sin(90) & \cos(90) \end{pmatrix} = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}. \tag{4.6}$$

A point $\mathbf{P} = (x, y)$ is therefore transformed to point $(y, -x)$. For a counterclockwise 90° rotation, (x, y) is transformed to $(-y, x)$. This is called the *negate and exchange* rule.

> Representations rotated not always by one hundred and eighty degrees, but sometimes by ninety or forty-five, completely subvert habitual perceptions of space; the outline of Europe, for instance, a shape familiar to anyone who has been even only to junior school, when swung around ninety degrees to the right, with the west at the top, begins to look like Denmark.
>
> —Georges Perec, *Life, A User's Manual* (1976).

The Golden Ratio

Start with a straight segment of length l and divide it into two parts a and b such that $a + b = l$ and $l/a = a/b$.



The ratio a/b is a constant called the *Golden Ratio* and is denoted ϕ. It is one of the important mathematical constants, like π and e, and was already known to the ancient Greeks. There seems to be a general belief that geometric figures can be made more pleasing to the eye if they obey this ratio. One example is the golden rectangle, whose sides are x and $x\phi$ long (Plate P.3). Many classical buildings and paintings seem to include this ratio. [Huntley 70] is a lively introduction to the Golden Ratio. It illustrates properties such as

$$\phi = \sqrt{1+\sqrt{1+\sqrt{1+\sqrt{1+\cdots}}}} \qquad \text{and} \qquad \phi = 1 + \frac{1}{1+\frac{1}{1+\frac{1}{\cdots}}}.$$

The value of ϕ is easy to determine. The basic ratio $l/a = a/b = \phi$ implies $(a+b)/a = a/b = \phi$, which, in turn, means $1 + b/a = \phi$ or $1 + 1/\phi = \phi$, an equation that can be written $\phi^2 - \phi - 1 = 0$. This equation is easy to solve, yielding $\phi = (1+\sqrt{5})/2 \approx 1.618\ldots$.

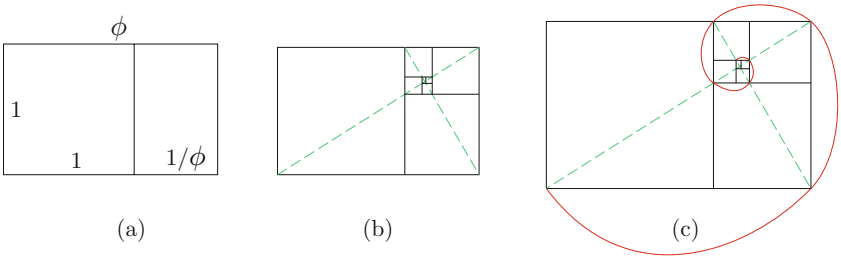


Figure 4.6: The Golden Ratio.

The equation $\phi = 1 + 1/\phi$ illustrates another unusual property of ϕ. Imagine the golden rectangle with sides 1 and ϕ (Figure 4.6a). Such a rectangle can be divided into a 1×1 square and a smaller golden rectangle of dimensions $1 \times 1/\phi$. The smaller rectangle can now be divided into a $1/\phi \times 1/\phi$ square and an even smaller golden rectangle (Figure 4.6b). When this process continues, the rectangles converge to a point. Figure 4.6c shows how a logarithmic spiral can be drawn through corresponding corners of the rectangles.

4.2.1 Homogeneous Coordinates

Unfortunately, our simple 2×2 transformation matrix cannot generate all the basic transformations that are needed in practice! In particular, it cannot generate *translation*. This is easy to see by arguing that any object containing the origin will, after any of the transformations above, still contain the origin [i.e., the result of $(0,0)\mathbf{T}$ is $(0,0)$ for any matrix $\mathbf{T}$].

Translations can be expressed by $x^* = x+m$, $y^* = y+n$, and one way to implement them is to generalize our transformations to $\mathbf{P}^* = \mathbf{PT} + (m,n)$, where $\mathbf{T}$ is the familiar 2×2 transformation matrix. A more elegant approach, however, is to stay with the compact notation $\mathbf{P}^* = \mathbf{PT}$ and to extend $\mathbf{T}$ to the 3×3 matrix

$$\mathbf{T} = \begin{pmatrix} a & b & 0 \\ c & d & 0 \\ m & n & 1 \end{pmatrix}. \tag{4.7}$$

This approach is called *homogeneous coordinates* and is commonly used in projective geometry. It makes it possible to unify all the two-dimensional transformations within one 3×3 matrix with six parameters. The problem is that a two-dimensional point (a pair) cannot be multiplied by a 3×3 matrix. This is solved by representing our points in homogeneous coordinates, which is done by extending the pair (x,y) to the triplet $(x,y,1)$. The rules for using homogeneous coordinates are the following:

1. To transform a point (x,y) to homogeneous coordinates, simply add a third component of 1. Hence, $(x,y) \Rightarrow (x,y,1)$.

2. To transform the triplet (a,b,c) from homogeneous coordinates back into a pair (x,y), divide by the third component. Hence, $(a,b,c) \Rightarrow (a/c, b/c)$.

This means that a point (x,y) has an infinite number of representations in homogeneous coordinates. Any triplet (ax,ay,a) where a is nonzero is a valid representation of the point. This suggests a way to intuitively understand homogeneous coordinates. We can consider the triplet (ax,ay,a) a point in three-dimensional space. When a varies from 0 to ∞, the point travels along a straight ray from the origin to infinity. The direction of the ray is determined by x and y but not by a. Therefore, each two-dimensional point (x,y) corresponds to a ray in three-dimensional space. To find the "real" location of the point, we look at the $z = 1$ plane. All points on this plane have coordinates $(x,y,1)$, so we only have to strip off the "1" in order to see where the point is located. Section 4.4 shows that homogeneous coordinates can also be applied to three-dimensional points.

◇ **Exercise 4.14:** Write the transformation matrix that performs (1) a y-reflection, (2) a translation by -1 in the x and y directions, and (3) a 180° counterclockwise rotation about the origin. Apply this compound transformation to the four corners $(1,1)$, $(1,-1)$, $(-1,1)$, and $(-1,-1)$ of a square centered on the origin. What are the transformed corners?

Matrix (4.7) is the general transformation matrix in two dimensions. It produces the most general linear transformation, $x^* = ax+cy+m$, $y^* = bx+dy+n$, and it shows that this transformation is fully specified by just six numbers.

We can gain a deeper understanding of homogeneous coordinates when we include two more parameters in matrix (4.7), writing it as

$$\begin{pmatrix} a & b & p \\ c & d & q \\ m & n & 1 \end{pmatrix}. \tag{4.8}$$

A general point (x, y) is now transformed to

$$(x, y, 1)\begin{pmatrix} a & b & p \\ c & d & q \\ m & n & 1 \end{pmatrix} = (ax + cy + m, bx + dy + n, px + qy + 1).$$

Applying rule 2 shows that the transformed point (x^*, y^*) is given by

$$x^* = \frac{ax + cy + m}{px + qy + 1}, \quad y^* = \frac{bx + dy + n}{px + qy + 1}.$$

To understand what this means, we apply this result to the four points $(2, 1)$, $(6, 1)$, $(2, 5)$, and $(6, 5)$ that constitute the four corners of a square (Figure 4.7a). Using the simple transformation

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

(i.e., no scaling, rotation, shearing, or translation and $p = q = 1$), the points are transformed to

$$\begin{aligned} \mathbf{P}_1 &= (2, 1) \rightarrow (2, 1, 4) \rightarrow (1/2, 1/4), \\ \mathbf{P}_2 &= (6, 1) \rightarrow (6, 1, 8) \rightarrow (3/4, 1/8), \\ \mathbf{P}_3 &= (2, 5) \rightarrow (2, 5, 8) \rightarrow (1/4, 5/8), \\ \mathbf{P}_4 &= (6, 5) \rightarrow (6, 5, 12) \rightarrow (1/2, 5/12). \end{aligned}$$

The transformed points (Figure 4.7b) also seem to form a square, but one that's viewed from a different direction and seen in perspective. This suggests that our transformation (using just p and q, without scaling, reflection, rotation, or shearing) has moved the square from its original position in the xy plane to another plane. Such transformations are called *projections* and are useful when dealing with objects in three-dimensional space.

4.2.2 Combining Transformations

Matrix notation is useful when working with transformations, because it makes it easy to combine transformations. To combine transformations $\mathbf{A}$, $\mathbf{B}$, and $\mathbf{C}$, we write the three transformation matrices and multiply them. An example is an x-reflection, followed by a y-scaling, followed by a 45° rotation

$$\begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix}\begin{pmatrix} 1 & 0 \\ 0 & 2 \end{pmatrix}\begin{pmatrix} 0.707 & -0.707 \\ 0.707 & 0.707 \end{pmatrix} = \begin{pmatrix} -0.707 & 0.707 \\ 1.414 & 1.414 \end{pmatrix}.$$

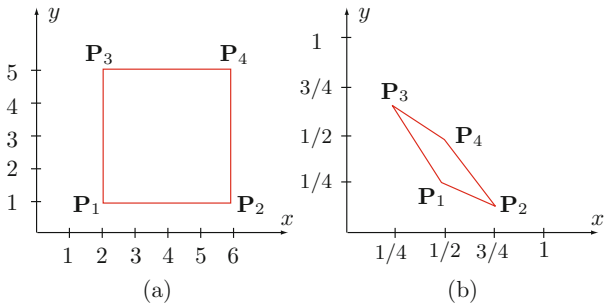


Figure 4.7: A Two-Dimensional Projection of a Square.

In general, matrix multiplication is noncommutative, reflecting the fact that geometric transformations are also noncommutative. It is easy to convince yourself that, for example, a rotation about the origin followed by a translation is not the same as a translation followed by a rotation about the origin.

Note that all the transformations discussed earlier are performed about the origin. Figure 4.8a shows an object rotated 40° clockwise. It is easy to see that the center of rotation is the origin. If, for example, we want to rotate an object about a point **P**, we have to translate both the object and the point such that **P** goes to the origin (Figure 4.8b), then rotate the object, and finally translate back (Figure 4.8c). Similarly, to reflect an object through an arbitrary line, we have to (1) translate the line (and the object) until it passes through the origin, (2) rotate the line (and the object) until it coincides with one of the coordinate axes, (3) reflect through that axis, (4) rotate back, and (5) translate back.

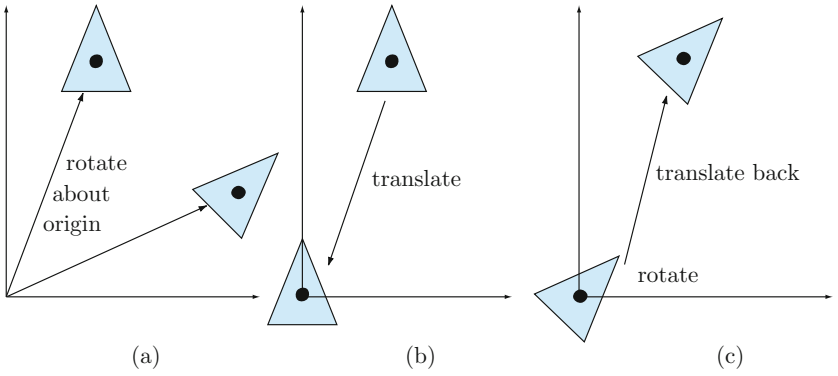


Figure 4.8: Rotation About a Point.

(Transformations are often done about the origin. See Exercise 6.11 for an example

on how this affects scaling in three dimensions.)

◇ **Exercise 4.15:** Derive the rotation matrix for a two-dimensional rotation about a point (x_0, y_0) using just trigonometry (i.e., without using translation).

Example: Reflection about the line $\mathbf{y} = \mathbf{x} + \mathbf{1}$. This line has a slope of 1 (i.e., it makes an angle of 45° with the x axis) and it intercepts the y axis at $y = 1$. We first translate down one unit, then rotate clockwise by 45°, then reflect through the x axis, rotate back, and translate back. The result is (α stands for both $\sin 45°$ and $\cos 45°$)

$$\begin{aligned}\mathbf{T} &= \begin{pmatrix}1 & 0 & 0\\ 0 & 1 & 0\\ 0 & -1 & 1\end{pmatrix}\begin{pmatrix}\alpha & -\alpha & 0\\ \alpha & \alpha & 0\\ 0 & 0 & 1\end{pmatrix}\begin{pmatrix}1 & 0 & 0\\ 0 & -1 & 0\\ 0 & 0 & 1\end{pmatrix}\begin{pmatrix}\alpha & \alpha & 0\\ -\alpha & \alpha & 0\\ 0 & 0 & 1\end{pmatrix}\begin{pmatrix}1 & 0 & 0\\ 0 & 1 & 0\\ 0 & 1 & 1\end{pmatrix}\\ &= \begin{pmatrix}0 & 2\alpha^2 & 1\\ 2\alpha^2 & 0 & 0\\ -2\alpha^2 & 1 & 1\end{pmatrix} = \begin{pmatrix}0 & 1 & 0\\ 1 & 0 & 0\\ -1 & 1 & 1\end{pmatrix}\end{aligned}$$

(because $2\alpha^2 = \sin^2 45° + \cos^2 45° = 1$). Note that $\det \mathbf{T} = -1$, i.e., pure reflection.

◇ **Exercise 4.16:** Demonstrate that the result in the example is correct.

Example: Reflection about an arbitrary line. Given the line $y = ax + b$, it is possible to reflect a point about this line by transforming the line to the x axis, reflecting about that axis, and transforming the line back. Since a is the slope (i.e., the tangent of the angle α between the line and the x axis) and b is the y intercept, the individual transformations needed are (1) a translation of $-b$ units in the y direction, (2) a clockwise rotation of α degrees about the origin, (3) a reflection about the x axis, (4) a counterclockwise rotation, and (5) a reverse translation. The combined transformation matrix is therefore

$$\begin{aligned}\mathbf{T}_{\text{reflect}} &= \begin{pmatrix}1 & 0 & 0\\ 0 & 1 & 0\\ 0 & -b & 1\end{pmatrix}\begin{pmatrix}\cos\alpha & -\sin\alpha & 0\\ \sin\alpha & \cos\alpha & 0\\ 0 & 0 & 1\end{pmatrix}\begin{pmatrix}1 & 0 & 0\\ 0 & -1 & 0\\ 0 & 0 & 1\end{pmatrix}\\ &\quad\times\begin{pmatrix}\cos\alpha & \sin\alpha & 0\\ -\sin\alpha & \cos\alpha & 0\\ 0 & 0 & 1\end{pmatrix}\begin{pmatrix}1 & 0 & 0\\ 0 & 1 & 0\\ 0 & b & 1\end{pmatrix}\\ &= \begin{pmatrix}\cos(2\alpha) & \sin(2\alpha) & 0\\ \sin(2\alpha) & -\cos(2\alpha) & 0\\ -b\sin(2\alpha) & 2b\cos^2\alpha & 1\end{pmatrix}. \qquad (4.9)\end{aligned}$$

The determinant of this transformation matrix equals -1, as should be for pure reflection. For the two special cases $\alpha = b = 0$ and $\alpha = 45°$ and $b = 0$, Equation (4.9) reduces to

$$\begin{pmatrix}1 & 0 & 0\\ 0 & -1 & 0\\ 0 & 0 & 1\end{pmatrix} \quad \text{and} \quad \begin{pmatrix}0 & 1 & 0\\ 1 & 0 & 0\\ 0 & 0 & 1\end{pmatrix}, \quad \text{respectively.}$$

One feature that makes Equation (4.9) less than general is the way the sine and cosine are obtained from the tangent of a known angle. Given that the slope a equals $\tan\alpha$, we can calculate

$$a = \tan\alpha = \frac{\sin\alpha}{\cos\alpha} = \frac{\sin\alpha}{\sqrt{1-\sin^2\alpha}},$$

which yields $\sin^2\alpha = a^2/(1+a^2)$ or

$$\sin\alpha = \pm\frac{a}{\sqrt{1+a^2}} \quad \text{and} \quad \cos\alpha = \pm\frac{1}{\sqrt{1+a^2}}.$$

The signs depend on the angle (or rather the quadrant in which the angle happens to be) and cannot be determined in a general way.

◇ **Exercise 4.17:** Compute the numerical value of matrix $\mathbf{T}_{\text{reflect}}$ for the case $\alpha = 30°$ and $b = 1$.

◇ **Exercise 4.18:** Digital images displayed on a screen or printed on paper consist of pixels. Even smooth curves are made of pixels. Thus, there is a need for efficient algorithms to compute the best pixels for a given curve or geometric figure. The circle has a high degree of symmetry, which is why it is possible to determine the best pixels for a given circle by computing the pixels for one octant and duplicating and transforming each pixel seven times to complete the remaining seven octants. The question is, is it possible to improve such an algorithm even more by doing half an octant and duplicating each pixel 15 times?

Another feature that makes Equation (4.9) less than general is the use of the explicit representation $y = ax + b$. This representation is limited because it cannot express vertical lines (for which a would be infinite). When reflecting a point about an arbitrary line, it is better to use the more general implicit representation of a straight line $ax + by + c = 0$, where a or b but not both can be zero. The slope of this line is $-a/b$, and substituting $b = 0$ yields a vertical line.

Given such a line, we start with a point $\mathbf{P} = (x, y)$ and its reflection $\mathbf{P}^* = (x^*, y^*)$ about the line. It is clear that the segment $\mathbf{PP}^*$ must be perpendicular to the line, so its equation must be $bx - ay + d = 0$. Since both $\mathbf{P}$ and $\mathbf{P}^*$ are on such a line, they satisfy $bx - ay + d = 0$ and $bx^* - ay^* + d = 0$. Subtracting these two expressions yields

$$b(x - x^*) = a(y - y^*). \tag{4.10}$$

We assume that $\mathbf{P}^*$ is the reflection of $\mathbf{P}$ about the line $ax + by + c = 0$, so the midpoint of segment $\mathbf{PP}^*$, which is the point $\big((x + x^*)/2, (y + y^*)/2\big)$, must be on this line and must therefore satisfy

$$a\frac{x + x^*}{2} + b\frac{y + y^*}{2} + c = 0. \tag{4.11}$$

Equations (4.10) and (4.11) can easily be solved for x^* and y^*. The solutions are

$$\mathbf{P}^* = (x^*, y^*) = \left(x - \frac{2a(ax + by + c)}{a^2 + b^2}, y - \frac{2b(ax + by + c)}{a^2 + b^2}\right)$$

$$= \left(\frac{(b^2-a^2)x - 2aby - 2ac}{a^2+b^2}, \frac{-2abx + (a^2-b^2)y - 2bc}{a^2+b^2} \right). \tag{4.12}$$

Equation (4.12) is easy to verify intuitively for vertical and for horizontal lines. When b is zero, the line becomes the vertical line $x = -c/a$ and Equation (4.12) reduces to

$$\mathbf{P}^* = (x^*, y^*) = \left(x - \frac{2a(ax+c)}{a^2}, y \right) = \left(-x - \frac{2c}{a}, y \right).$$

When $a = 0$, the line is the horizontal $y = -c/b$, and the same equation reduces to

$$\mathbf{P}^* = (x^*, y^*) = \left(x, y - \frac{2b(by+c)}{b^2} \right) = \left(x, -y - \frac{2c}{b} \right).$$

The transformation matrix for reflection about an arbitrary line $ax + by + c = 0$ is directly obtained from Equation (4.12)

$$\mathbf{T} = \begin{pmatrix} b^2-a^2 & -2ab & 0 \\ -2ab & a^2-b^2 & 0 \\ -2ac & -2bc & \frac{1}{a^2+b^2} \end{pmatrix}. \tag{4.13}$$

Its determinant is

$$\det \mathbf{T} = \frac{(b^2-a^2)(a^2-b^2) - 4a^2b^2}{a^2+b^2} = -\frac{a^4 + 2a^2b^2 + b^4}{a^2+b^2} = -(a^2+b^2),$$

which equals -1 (pure reflection) for lines expressed in the standard form (defined as the case where $a^2 + b^2 = 1$).

◇ **Exercise 4.19:** Use Equation (4.12) to obtain the transformation rule for reflection about a line that passes through the origin.

We turn now to the product of two reflections about the two arbitrary lines L_1: $ax + by + c = 0$ and L_2: $dx + ey + f = 0$ (Figure 4.9a). This product can be calculated from Equation (4.13) as the matrix product

$$\begin{pmatrix} b^2-a^2 & -2ab & 0 \\ -2ab & a^2-b^2 & 0 \\ -2ac & -2bc & \frac{1}{a^2+b^2} \end{pmatrix} \begin{pmatrix} e^2-d^2 & -2de & 0 \\ -2de & d^2-e^2 & 0 \\ -2df & -2ef & \frac{1}{d^2+e^2} \end{pmatrix},$$

but this product is complex and difficult to interpret geometrically. In order to simplify it, we assume, without loss of generality, that both lines pass through the origin and that the first is also horizontal (Figure 4.9b). The first assumption means that the lines intersect at the origin and that $c = f = 0$. The second assumption means that the first line is identical to the x axis, so $a = 0$ and $b = 1$. Also, $f = 0$ implies $dx + ey = 0$ or $y = -(d/e)x$. The quantity $-d/e$ is the slope (i.e., $\tan\theta$) of the second line, so we conclude that

$$-\frac{d}{e} = -\tan\theta = -\frac{\sin\theta}{\cos\theta}, \quad \text{implying } d^2 + e^2 = 1.$$

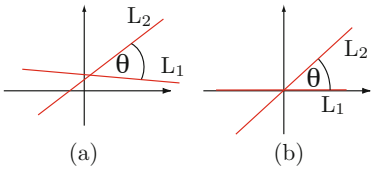


Figure 4.9: Reflections About Two Intersecting Lines.

Under these assumptions, the matrix product above becomes

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} e^2-d^2 & -2de & 0 \\ -2de & d^2-e^2 & 0 \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} e^2-d^2 & -2de & 0 \\ 2de & e^2-d^2 & 0 \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} \cos(2\theta) & -\sin(2\theta) & 0 \\ \sin(2\theta) & \cos(2\theta) & 0 \\ 0 & 0 & 1 \end{pmatrix}, \tag{4.14}$$

leading to the important conclusion that the product of two reflections about arbitrary lines is a rotation through an angle 2θ about the intersection point of the lines, where θ is the angle between the lines. It can be shown that the opposite is also true; any rotation is the product of two reflections about two intersecting lines.

The discussion above assumes that both lines pass through the origin. In the special case where $\theta = 0$, such lines would be identical, so reflecting a point $\mathbf{P}$ about them would move it back to itself. However, for $\theta = 0$, matrix (4.14) reduces to the identity matrix, so it is valid even for identical lines.

In the special case where the lines are parallel, their intersection point is at infinity and a rotation about a center at infinity is a translation.

◇ **Exercise 4.20:** Given the two parallel lines $y = 0$ and $y = c$, calculate the double reflection of a point (x, y) about them.

◇ **Exercise 4.21:** Consider the shearing transformation $\mathbf{T}_a$ of Equation (4.15), followed by the 90° rotation $\mathbf{T}_b$. What is the combined transformation, and what kind of transformation is it?

$$\mathbf{T}_a = \begin{pmatrix} 0 & 1 & 0 \\ 2 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \qquad \mathbf{T}_b = \begin{pmatrix} \cos 90^\circ & -\sin 90^\circ & 0 \\ \sin 90^\circ & \cos 90^\circ & 0 \\ 0 & 0 & 1 \end{pmatrix}. \tag{4.15}$$

◇ **Exercise 4.22:** Given the two rotations

$$\mathbf{T}_1 = \begin{pmatrix} \cos\theta_1 & -\sin\theta_1 & 0 \\ \sin\theta_1 & \cos\theta_1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad \text{and} \quad \mathbf{T}_2 = \begin{pmatrix} \cos\theta_2 & -\sin\theta_2 & 0 \\ \sin\theta_2 & \cos\theta_2 & 0 \\ 0 & 0 & 1 \end{pmatrix},$$

calculate the combined transformation $\mathbf{T}_1\mathbf{T}_2$. Is it identical to a rotation through $(\theta_1 + \theta_2)$?

◇ **Exercise 4.23:** Given the two shearing transformations

$$\mathbf{T}_1 = \begin{pmatrix} 1 & b & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad \text{and} \quad \mathbf{T}_2 = \begin{pmatrix} 1 & 0 & 0 \\ c & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix},$$

calculate the combined transformation $\mathbf{T}_1\mathbf{T}_2$. Is it identical to a shearing by factors b and c?

◇ **Exercise 4.24:** Prove that three successive shearings about the x, y, and x axes is equivalent to a rotation about the origin.

◇ **Exercise 4.25:** Matrix $\left(\begin{smallmatrix} a & 0 \\ 0 & d \end{smallmatrix}\right)$ scales an object by factors a and d along the x and y axes, respectively. If we want to scale the object by the same factors, but in the i and j directions (see Figure 4.10, where i and j are perpendicular and form an angle θ with the x and y axes, respectively), we need to (1) rotate the object θ degrees clockwise, (2) scale along the x and y axes using matrix $\left(\begin{smallmatrix} a & 0 \\ 0 & d \end{smallmatrix}\right)$, and (3) rotate back. Write the three transformation matrices and their product. Discuss the case $a = d$ (uniform scaling).

◇ **Exercise 4.26:** We can perform an exercise with shearing, similar to Exercise 4.25. Matrix $\left(\begin{smallmatrix} 1 & b \\ c & 1 \end{smallmatrix}\right)$ shears an object by factors c and b along the x and y axes, respectively. Calculate the matrix that shears the object by the same factors, but in the i and j directions (Figure 4.10).

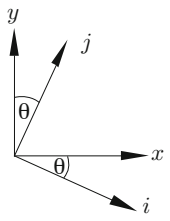


Figure 4.10: Scaling Along Rotated Axes.

◇ **Exercise 4.27:** Discuss scaling relative to a point (x_0, y_0), and show that the result is identical to the product of a translation followed by scaling, followed by a reverse translation.

Using Equation (Ans.2) in the Answers to Exercises, it is easy to explore the effect of two consecutive scaling transformations, with scaling factors of k_1 and k_2 and about points $\mathbf{P}_1 = (x_1, y_1)$ and $\mathbf{P}_2 = (x_2, y_2)$, respectively. We simply multiply the two transformation matrices

$$\begin{pmatrix} k_1 & 0 & 0 \\ 0 & k_1 & 0 \\ x_1(1-k_1) & y_1(1-k_1) & 1 \end{pmatrix} \begin{pmatrix} k_2 & 0 & 0 \\ 0 & k_2 & 0 \\ x_2(1-k_2) & y_2(1-k_2) & 1 \end{pmatrix}$$
$$= \begin{pmatrix} k_1k_2 & 0 & 0 \\ 0 & k_1k_2 & 0 \\ k_2(1-k_1)x_1 + (1-k_2)x_2 & k_2(1-k_1)y_1 + (1-k_2)y_2 & 1 \end{pmatrix}. \qquad (4.16)$$

The result is similar to Equation (Ans.2) except for the bottom row. It seems that the product of two scalings is a third scaling with a factor k_1k_2, but about what point? To write Equation (4.16) in the form of Equation (Ans.2), we write

$$k_2(1-k_1)x_1 + (1-k_2)x_2 = x_c(1-k_1k_2),$$
$$k_2(1-k_1)y_1 + (1-k_2)y_2 = y_c(1-k_1k_2),$$

and solve for (x_c, y_c), obtaining

$$x_c = \frac{k_2(1-k_1)x_1 + (1-k_2)x_2}{1-k_1k_2},$$
$$y_c = \frac{k_2(1-k_1)y_1 + (1-k_2)y_2}{1-k_1k_2}.$$

The center of the double scaling is therefore point

$$\mathbf{P}_c = \frac{k_2(1-k_1)}{1-k_1k_2}\mathbf{P}_1 + \frac{1-k_2}{1-k_1k_2}\mathbf{P}_2 = a\mathbf{P}_1 + b\mathbf{P}_2.$$

Notice that $a + b = 1$, which is why $\mathbf{P}_c$ is a point on the straight segment connecting $\mathbf{P}_1$ and $\mathbf{P}_2$ (see also Equation (Ans.42)).

In the special case $\mathbf{P}_1 = \mathbf{P}_2$, it is easy to see that the center of the double scaling is $\mathbf{P}_c = \mathbf{P}_1 = \mathbf{P}_2$.

◇ **Exercise 4.28:** What is the result of two consecutive scalings with the same scaling factors but about two different points?

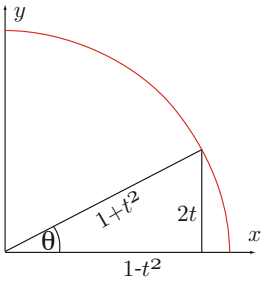


Figure 4.11: A Unit Circle.

◇ **Exercise 4.29:** Show that all the points with coordinates (t^2, t), where $0 \le t \le 1$, after being transformed by

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

lie on the perimeter of the unit circle $x^2 + y^2 = 1$. (Hint: See Figure 4.11.)

It is easy to see that the transformations discussed here can change lengths and angles. Scaling changes the lengths of objects. Rotation and shearing change angles. One feature that's preserved, though, is parallel lines. A pair of parallel lines will remain parallel after any scaling, reflection, rotation, shearing, and translation. A transformation that preserves parallelism (and also maps finite points to finite points) is called *affine*.

4.2.3 Fast Rotations

Rotation requires the calculation of the transcendental sine and cosine functions, which is time consuming. If many rotations are needed, it is preferable to precompute the trigonometric functions for many angles and store them in a table. This section shows how to do this using integers only, a method that results in much faster rotations than using floating-point numbers.

The method is illustrated for the first quadrant (rotation angles of 0° to 90°) in increments of 1°. Notice that rotations in other quadrants can be achieved by a first-quadrant rotation followed by a reflection. The following *Mathematica* code generates 91 sine values, from $\sin 0° = 0$ to $\sin 90° = 1$, multiplies each by $2^{14} = 16{,}384$, rounds them, and stores them in a table as 16-bit integers ranging from 0 to 16,384.

```
d2r=Pi/180;
Table[Round[N[16384*Sin[i*d2r]]], {i,0,90}]
```

The 91 values are listed in Table 4.12, but notice that they are only approximations of the true sine values. (Even floating-point sine values are, in general, just approximations, but normally better than our integers.) This means that the use of this table for many successive rotations of a point may place it farther and farther away from its true position. When we perform many successive rotations of an object that consists

θ	$\sin\theta$	θ	$\sin\theta$	θ	$\sin\theta$	θ	$\sin\theta$	θ	$\sin\theta$
0	0	1	286	2	572	3	857	4	1143
5	1428	6	1713	7	1997	8	2280	9	2563
10	2845	11	3126	12	3406	13	3686	14	3964
15	4240	16	4516	17	4790	18	5063	19	5334
20	5604	21	5872	22	6138	23	6402	24	6664
25	6924	26	7182	27	7438	28	7692	29	7943
30	8192	31	8438	32	8682	33	8923	34	9162
35	9397	36	9630	37	9860	38	10087	39	10311
40	10531	41	10749	42	10963	43	11174	44	11381
45	11585	46	11786	47	11982	48	12176	49	12365
50	12551	51	12733	52	12911	53	13085	54	13255
55	13421	56	13583	57	13741	58	13894	59	14044
60	14189	61	14330	62	14466	63	14598	64	14726
65	14849	66	14968	67	15082	68	15191	69	15296
70	15396	71	15491	72	15582	73	15668	74	15749
75	15826	76	15897	77	15964	78	16026	79	16083
80	16135	81	16182	82	16225	83	16262	84	16294
85	16322	86	16344	87	16362	88	16374	89	16382
90	16384								

Table 4.12: Sine Values as 16-Bit Integers.

of many points, placing points away from where they should be generally results in a deformation of the object.

We assume that the points are represented by coordinates that are 16-bit integers. Calculating the rotated coordinates (x^*, y^*) of a point (x, y) can now be done, for example, by

$$x^* = \text{rshift}(x \times \text{Table}(90-\theta), 14) - \text{rshift}(y \times \text{Table}(\theta), 14),$$
$$y^* = \text{rshift}(x \times \text{Table}(\theta), 14) + \text{rshift}(y \times \text{Table}(90-\theta), 14).$$

Notice how the required cosine values are obtained from the end of the table. This method works because the table has 91 entries. Multiplying a 16-bit integer coordinate by a 16-bit integer sine value yields a 32-bit product. The right shift effectively divides the product by $2^{14} = 16{,}384$, a necessary operation because our integer sine values have been premultiplied by this scale factor.

◇ **Exercise 4.30:** Use this method to calculate the results of rotating point $(1, 2)$ by $60°$ and by $80°$. In each case, compare the results with those obtained when built-in sine and cosine functions are used.

4.2.4 CORDIC Rotations

We routinely use calculators to compute values of common functions, but have you ever wondered how a calculator determines the value of, say, $\tan 72.81°$ so fast? Many calculators use CORDIC (COordinate Rotation, DIgital Computer), a general method for

computing many elementary functions. CORDIC was originally proposed by [Volder 59] and was extended by [Walther 71]. The original references are hard to find but are included in [Swartzlander 90]. Here, we show how CORDIC can be used to implement fast rotations.

It is sufficient to consider a rotation about the origin where the rotation angle θ is in the interval $[0, 90°)$ (the first quadrant). The special case $\theta = 90°$ can be implemented by the negate and exchange rule, Equation (4.6). Rotations in other quadrants can be achieved by a first-quadrant rotation, followed by a reflection.

The rotation is expressed by [see Equation (4.4)]

$$(x^*, y^*) = (x, y) \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix}. \tag{4.17}$$

Because θ is less than 90°, we know that $\cos\theta$ is nonzero, so we can factor out $\cos\theta$, yielding

$$(x^*, y^*) = \cos\theta\,(x, y) \begin{pmatrix} 1 & -\tan\theta \\ \tan\theta & 1 \end{pmatrix}.$$

We now express θ as the sum $\sum_{i=0}^{m} \theta_i$, where angles θ_i are defined by the relation $\tan\theta_i = 2^{-i}$ or $\theta_i \stackrel{\text{def}}{=} \arctan(2^{-i})$. The first 16 θ_i, for $i = 0, 1, \ldots, 15$, are listed in Table 4.13.

i	θ_i (degrees)	θ_i(radians)	K_i
0	45.	0.785398	0.70710678118654746
1	26.5651	0.463648	0.63245553203367577
2	14.0362	0.244979	0.61357199107789628
3	7.12502	0.124355	0.60883391251775243
4	3.57633	0.0624188	0.60764825625616825
5	1.78991	0.0312398	0.60735177014129604
6	0.895174	0.0156237	0.60727764409352614
7	0.447614	0.00781234	0.60725911229889284
8	0.223811	0.00390623	0.60725447933256249
9	0.111906	0.00195312	0.60725332108987529
10	0.0559529	0.000976562	0.60725303152913446
11	0.0279765	0.000488281	0.60725295913894495
12	0.0139882	0.000244141	0.60725294104139727
13	0.00699411	0.00012207	0.60725293651701029
14	0.00349706	0.0000610352	0.60725293538591352
15	0.00174853	0.0000305176	0.60725293510313938

Table 4.13: The First 16 θ_i's and Scale Factors.

In order to express any angle θ as the sum of these particular θ_i, some θ_i will have to be subtracted. Consider, for example, $\theta = 58°$. We start with $\theta_0 = 45°$. Since $\theta_0 < \theta$, we add θ_1. The sum $\theta_0 + \theta_1 = 45 + 26.5651 = 71.5651$ is greater than θ, so we subtract θ_2. The new sum, 57.5289, is less than θ, so we add θ_3, and so on.

◇ **Exercise 4.31:** We want to be able to express any angle θ in the range $[0°, 90°)$ by adding and subtracting a number of consecutive θ_i, from θ_0 to some θ_m, without skipping any θ_i in between. Is that possible?

It is easy to write a program that decides which of the θ_i's should be added and which should be subtracted. Thus, we end up with

$$\theta = \sum_{i=0}^{m} d_i\theta_i = \sum_{i=0}^{m} d_i \arctan(2^{-i}), \quad \text{where} \quad d_i = \pm 1.$$

Once the number m of necessary d_i's and their values have been determined, we rotate (x, y) to (x^*, y^*) in a loop where each iteration rotates a point (x_i, y_i) through an angle $d_i\theta_i$ to a point (x_{i+1}, y_{i+1}). A general iteration can be expressed in the form

$$\begin{aligned}(x_{i+1}, y_{i+1}) &= \cos(d_i\theta_i)\,(x_i, y_i)\begin{pmatrix} 1 & -d_i\tan\theta_i \\ d_i\tan\theta_i & 1 \end{pmatrix} \\ &= \cos(d_i\theta_i)\,(x_i, y_i)\begin{pmatrix} 1 & -d_i 2^{-i} \\ d_i 2^{-i} & 1 \end{pmatrix} \\ &= \cos(d_i\theta_i)\,(x_i + y_i d_i 2^{-i}, y_i - x_i d_i 2^{-i}). \end{aligned} \tag{4.18}$$

We interpret the result (x_{i+1}, y_{i+1}) of an iteration as the vector from the origin to point (x_{i+1}, y_{i+1}). Equation (4.18) shows that this vector is the product of two terms. The second term, $(x_i + y_i d_i 2^{-i}, y_i - x_i d_i 2^{-i})$, determines the direction of the vector, while the first term, $\cos(d_i\theta_i)$, affects only the magnitude of the vector. The second term is easy to calculate since it just involves shifts. We know that d_i is just a sign and that a product of the form $x_i 2^{-i}$ can be computed by shifting x_i i positions to the right. The problem is to calculate the first term, $\cos(d_i\theta_i)$, and to multiply the two terms.

This is why CORDIC proceeds by first performing all the iterations

$$(x_{i+1}, y_{i+1}) \leftarrow (x_i + y_i d_i 2^{-i}, y_i - x_i d_i 2^{-i})$$

using just right shifts and additions/subtractions; the cosine terms are ignored. The result is a vector that points in the right direction but is too long (Figure 4.14). To bring this vector to its correct size, it should be multiplied by the scale factor

$$K_m = \prod_{i=0}^{m} \cos\theta_i.$$

(Notice that $\cos(d_i\theta_i) = \cos\theta_i$ since cosine is an even function.) This is discouraging because it suggests that m multiplications are needed just to calculate the scale factor K_m. However, the first 16 scale factors are listed in Table 4.13 and even a quick glance shows that they converge to the number 0.60725.... Reference [Vachss 87] shows that K_m can be obtained simply by using the m most significant bits of this number and ignoring the rest.

Using the identity $\sin^2\theta + \cos^2\theta = 1$ and the definition $\tan\theta_i = 2^{-i}$, we get

$$\cos\theta_i = \frac{1}{\sqrt{1+\tan^2\theta_i}} = \frac{1}{\sqrt{1+2^{-2i}}},$$

which is why the scale factors of Table 4.13 were so easily calculated to a high precision by the code

```
N[Table[Product[(2^(-2i)+1)^(-1/2),{i,0,n}],{n,0,16}],17]//TableForm
```

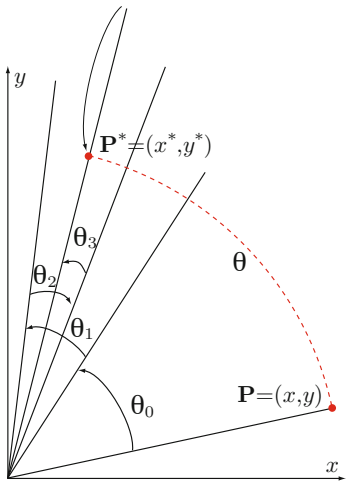


Figure 4.14: CORDIC Rotation.

◇ **Exercise 4.32:** Suggest another way to calculate K_m.

Any practical CORDIC implementation (see [Jarvis 90] for a C program) should have the following two features.

1. CORDIC employs only shifts and additions/subtractions, so any implementation should use fixed-point, instead of floating-point, arithmetic. This is fast since shifting and adding fixed-point numbers can be done with integer operations. Notice that all the numbers involved in the computations are less than unity, except perhaps the original coordinates (x, y). A software package for graphics employing this method should therefore use normalized coordinates (fixed-point numbers in the interval $[0, 1]$) throughout and perform all the calculations on these small numbers. Each iteration results in a pair (x_{i+1}, y_{i+1}) that's slightly larger than its predecessor, but the last iteration results in a pair that can be larger than (x, y) by a factor of at most $1/0.60725\ldots = 1.64676\ldots$. This pair is then scaled down when multiplied by K_m. The final step is to scale the final coordinates up.

All this suggests a 32-bit fixed-point format where the leftmost bit is reserved, as usual, for the sign, the next two bits are the integer part, and the remaining 29 bits are

the fractional part (29 bits being equivalent to 9 decimal digits). The largest number that can be represented by this format is $11.11\ldots1_2 = 3.999\ldots$ and the smallest one is $110\ldots0_2 = -4$. It's a good idea to reserve two bits for the integer part because (1) even though all the numbers involved are 1 or smaller, some intermediate results may be greater than 1 and (2) this convention makes it possible to represent the important constants π, e, and ϕ (the Golden Ratio).

2. Earlier, we said, "It is easy to write a program that decides which of the θ_i's should be added and which should be subtracted." The practical way to do this is to initialize a variable z to θ and try to drive z to zero during the iterations. In iteration i the program should calculate both $z+\theta_i$ and $z-\theta_i$, select the value that's closer to zero, use it to decide whether to add or subtract θ_i, and then update z. If $z-\theta_i$ is closer to zero, then θ_i should be added; otherwise, θ_i should be subtracted. An example is $\theta = 58°$. We initialize z to 58. In iteration 0, it is clear that $58-45 = 13$ is closer to zero than $58+45$. The program therefore adds θ_0 and updates z to 13. In iteration 1, the program finds that $13-26.5651 = -13.5651$ is closer to zero than $13+26.5651$, so it adds θ_1 and updates z to -13.5651. In iteration 2, the program discovers that $-13.5651+14.0362 = 0.4711$ is closer to zero than $-13.5651-14.0362$, so it subtracts θ_2 and updates z to 0.4711.

Finally, we realize that there is really no need to compare $z+\theta_i$ and $z-\theta_i$ in iteration i. We simply start by selecting $d_0 = +1$ and update z by subtracting $z \leftarrow z-\theta_0$, $z \leftarrow z-\theta_1$, etc., until we get a negative value in z. We then change d_i to -1 (the new sign of z) and update z by $z \leftarrow z-d_i\theta_i$ (which now amounts to adding θ_i to z). This is summarized by the *Mathematica* code of Figure 4.15. (But note that the `Sign` function of *Mathematica* returns $+1$, 0, or -1, while we need a result of $+1$ or -1. The code as shown is simple but not completely general.)

```
t=Table[ArcTan[2.^{-i}], {i,0,15}]; (* arctans in radians *)
d=1; x=2.1; y=0.34; z=46. Degree;
Do[{Print[i,", ",x,", ",y,", ",z,", ",d],
 xn=x+y d 2^{-i}, yn=y-x d 2^{-i},
 zn=z-d t[[i+1]], d=Sign[zn], x=xn, y=yn, z=zn}, {i,0,14}]
Print[0.60725x,", ",0.60725y]
```

Figure 4.15: *Mathematica* Code for CORDIC Rotations.

> Compared to other approaches, CORDIC is a clear winner when a hardware multiplier is unavailable (e.g. in a microcontroller) or when you want to save the gates required to implement one (e.g. in an FPGA). On the other hand, when a hardware multiplier is available (e.g. in a DSP microprocessor), table-lookup methods and good old-fashioned power series are generally faster than CORDIC.
>
> —Grant R. Griffin, www.dspguru.com/info/faqs/cordic.htm

◇ **Exercise 4.33:** Instead of using the complex CORDIC method, wouldn't it be simpler to perform a rotation by a direct use of Equation (4.17)? After all, this only requires the calculation of one sine and one cosine values.

4.2.5 Similarities

A *similarity* is a transformation that scales all distances by a fixed factor. It is easy to show that a similarity is produced by the special transformation matrix

$$\begin{pmatrix} a & c & 0 \\ -c & a & 0 \\ m & n & 1 \end{pmatrix}.$$

To show this, we observe that translations preserve distances, so we can ignore the translation part of the matrix above and restrict ourselves to the matrix $\left(\begin{smallmatrix} a & c \\ -c & a \end{smallmatrix}\right)$. It transforms a point $\mathbf{P} = (x, y)$ to the point $\mathbf{P}^* = (x^*, y^*) = (ax - cy, cx + ay)$. Given the two transformations $\mathbf{P}_1 \to \mathbf{P}_1^*$ and $\mathbf{P}_2 \to \mathbf{P}_2^*$, it is straightforward to illustrate the relation

$$\begin{aligned} \texttt{distance}^2(\mathbf{P}_1^*\mathbf{P}_2^*) &= \left((\Delta x^*)^2 + (\Delta y^*)^2\right) \\ &= [(ax_2 - cy_2) - (ax_1 - cy_1)]^2 + [(cx_2 + ay_2) - (cx_1 + ay_1)]^2 \\ &= (a\Delta x - c\Delta y)^2 + (c\Delta x + a\Delta y)^2 \\ &= a^2\Delta x^2 - 2a\Delta x c\Delta y + c^2\Delta y^2 + c^2\Delta x^2 + 2c\Delta x a\Delta y + a^2\Delta y^2 \\ &= (a^2 + c^2)(\Delta x^2 + \Delta y^2) \\ &= (a^2 + c^2)\texttt{distance}^2(\mathbf{P}_1\mathbf{P}_2), \end{aligned}$$

implying that $\texttt{distance}(\mathbf{P}_1^*\mathbf{P}_2^*) = \sqrt{a^2 + c^2}\,\texttt{distance}(\mathbf{P}_1\mathbf{P}_2)$. Thus, all distances are scaled by a factor of $\sqrt{a^2 + c^2}$.

In general, a similarity is a transformation of the form $\mathbf{P}^* = (x^*, y^*) = (ax - cy + m, \pm(cx + ay) + n)$, where the ratio of expansion (or shrinking) is $k = \sqrt{a^2 + c^2}$. If k is positive, the similarity is called *direct*; if k is negative, the similarity is *opposite*.

◇ **Exercise 4.34:** Discuss the case $k = 0$.

Using the ratio k, we can write a similarity (ignoring the translation part) as the product

$$\begin{pmatrix} a & c & 0 \\ -c & a & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} k & 0 & 0 \\ 0 & k & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} a/k & c/k & 0 \\ -c/k & a/k & 0 \\ 0 & 0 & 1 \end{pmatrix},$$

which shows that a similarity is a combination of a scaling/reflection (by a factor k) and a rotation. (The definition of k implies that $(a/k)^2 + (c/k)^2 = 1$, so we can consider c/k and a/k the sine and cosine of the rotation angle, respectively.)

4.2.6 A 180° Rotation

Another interesting example of combining transformations is a 180° rotation about a fixed point $\mathbf{P} = (P_x, P_y)$. This combination is called a *halfturn*. It is performed, as usual, by translating $\mathbf{P}$ to the origin, rotating about the origin, and translating back.

The transformation matrix is (notice that $\cos(180°) = -1$)

$$\mathbf{T} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -P_x & -P_y & 1 \end{pmatrix} \begin{pmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ P_x & P_y & 1 \end{pmatrix} = \begin{pmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 2P_x & 2P_y & 1 \end{pmatrix}.$$

A general point (x, y) is therefore transformed by a halfturn to

$$(x, y, 1) \begin{pmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 2P_x & 2P_y & 1 \end{pmatrix} = (-x + 2P_x, -y + 2P_y, 1) \tag{4.19}$$

(Figure 4.16a), but it's more interesting to explore the effect of two consecutive halfturns, about points **P** and **Q**. The second halfturn transforms point $(-x + 2P_x, -y + 2P_y, 1)$ to

$$(-x+2P_x, -y+2P_y, 1) \begin{pmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 2Q_x & 2Q_y & 1 \end{pmatrix} = (x - 2P_x + 2Q_x, y - 2P_y + 2Q_y, 1). \tag{4.20}$$

If $\mathbf{P} = \mathbf{Q}$, then the result of the second halfturn is (x, y), showing how two identical 180° rotations return a point to its original location. If **P** and **Q** are different, the result is a *translation* of the original point (x, y) by factors $-2P_x + 2Q_x$ and $-2P_y + 2Q_y$ (Figure 4.16b).

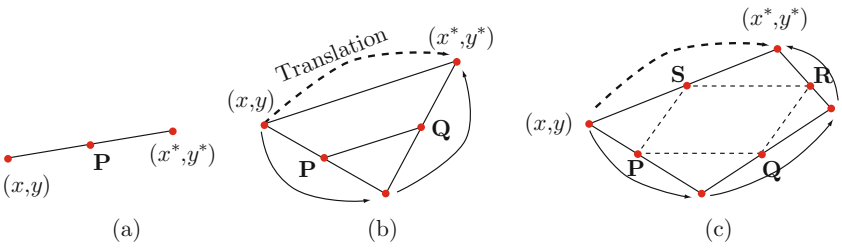


Figure 4.16: Halfturns.

◇ **Exercise 4.35:** What is the result of three consecutive halfturns about the distinct points **P**, **Q**, and **R**?

> Things turn out best for the people who make the best out of the way things turn out.
>
> —Art Linkletter.

4.2.7 Glide Reflections

This transformation is a special combination of three reflections. Imagine the two vertical parallel lines $x = L$ and $x = M$ and the horizontal line $y = N$ (Figure 4.17a). Reflecting a point $\mathbf{P} = (x, y)$ about the line $x = L$ is done by translating the line to the y axis, reflecting about that axis, and translating back. The transformation matrix is

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -L & 0 & 1 \end{pmatrix} \begin{pmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ L & 0 & 1 \end{pmatrix} = \begin{pmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 2L & 0 & 1 \end{pmatrix},$$

and the transformed point is

$$(x, y, 1) \begin{pmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 2L & 0 & 1 \end{pmatrix} = (-x + 2L, y, 1).$$

Reflecting this point about the line $x = M$ results in

$$(-x + 2L, y, 1) \begin{pmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 2M & 0 & 1 \end{pmatrix} = (x - 2L + 2M, y, 1)$$

(a translation), and reflecting this about the horizontal line $y = N$ yields

$$(x - 2L + 2M, y, 1) \begin{pmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 2N & 1 \end{pmatrix} = (x - 2L + 2M, -y + 2N, 1).$$

This particular glide reflection is therefore a translation in x and a reflection in y. A general glide reflection is the product of three reflections, the first two about parallel lines L and M and the third about a line N perpendicular to them (Figure 4.17b).

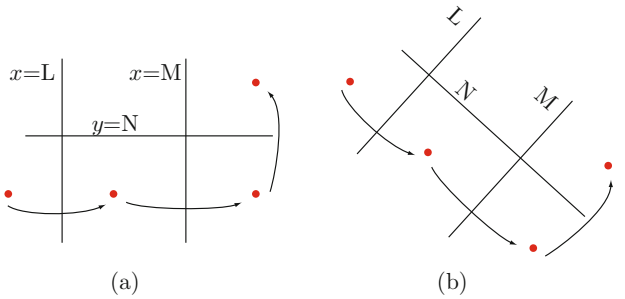


Figure 4.17: Glide Reflection.

4.2.8 Improper Rotations

A rotation followed by a reflection about one of the coordinate axes is called an *improper rotation*. The transformation matrices for the two possible improper rotations in two dimensions (Figure 4.18) are

$$\begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} = \begin{pmatrix} \cos\theta & \sin\theta \\ \sin\theta & -\cos\theta \end{pmatrix},$$
$$\begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} -\cos\theta & -\sin\theta \\ -\sin\theta & \cos\theta \end{pmatrix},$$

and the transformation rules therefore are

$$x^* = x\cos\theta + y\sin\theta, \quad y^* = x\sin\theta - y\cos\theta,$$
$$x^* = -x\cos\theta - y\sin\theta, \quad y^* = -x\sin\theta + y\cos\theta.$$

Notice that the determinant of an improper rotation matrix equals -1, like that of a pure reflection.

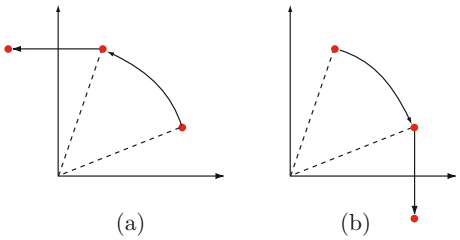


Figure 4.18: Improper Rotations.

An improper rotation differs from a rotation in one important aspect. When we rotate an object through a small angle and repeat this transformation, the object seems to move smoothly along a circle. Each time we repeat an improper rotation, however, the object "jumps" from one side of the coordinate plane to the other. The total effect is very different from that of a smooth circular movement.

4.2.9 Decomposing Transformations

Sometimes, a certain transformation A may be equivalent to the combined effects of several different transformations B, C, and D. We say that A can be *decomposed* into B, C, and D. Mathematically, this is equivalent to saying that the original transformation matrix $\mathbf{T}_A$ equals the product $\mathbf{T}_B\mathbf{T}_C\mathbf{T}_D$. We have already seen that a rotation in two dimensions can be decomposed into a scaling followed by a shearing; here are other examples.

It may come as a surprise that the general two-dimensional transformation matrix, Equation (4.7), can be written as a product of shearing, scaling, rotation, and translation

as follows:

$$\begin{bmatrix} a & b & 0 \\ c & d & 0 \\ m & n & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ (ac+bd)/A^2 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} A & 0 & 0 \\ 0 & (ad-bc)/A & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} a/A & b/A & 0 \\ -b/A & a/A & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ m & n & 1 \end{bmatrix}, \tag{4.21}$$

where $A = \sqrt{a^2 + b^2}$. The third matrix produces rotation since $(a/A)^2 + (b/A)^2 = 1$.

Even something as simple as shearing in one direction can be written as the product of a unit shearing and two scalings:

$$\begin{pmatrix} 1 & 0 & 0 \\ c & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 1/c & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} c & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

Even the simple transformation of a unit shearing can be decomposed into a product that involves a scaling and two rotations. Note that the Golden Ratio ϕ is involved,

$$\begin{pmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} \cos\alpha & -\sin\alpha & 0 \\ \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \phi & 0 & 0 \\ 0 & 1/\phi & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \cos\beta & \sin\beta & 0 \\ -\sin\beta & \cos\beta & 0 \\ 0 & 0 & 1 \end{pmatrix},$$

where $\alpha = \tan^{-1}\phi \approx 58.28°$ and $\beta = \tan^{-1}(1/\phi) \approx 31.72°$.

(This is indeed a surprising result. It means that a clockwise rotation of 58.28°, followed by a scaling of ϕ in the x direction and $1/\phi$ in the y direction, followed by a counterclockwise rotation of 31.72°, is equivalent to a unit shear in the x direction. This is illustrated by Figure 4.19.)

> Geometry has two great treasures: one the Theorem of Pythagoras; the other, the division of a line into extreme and mean ratio. The first we may compare to a measure of gold; the second we may name a precious jewel.
>
> —Johannes Kepler.

◇ **Exercise 4.36:** Given the transformation

$$x^* = 3x - 2y + 1, \quad y^* = 4x + 5y - 6,$$

calculate the transformation matrix and decompose it into a product of four matrices as shown in Equation (4.21).

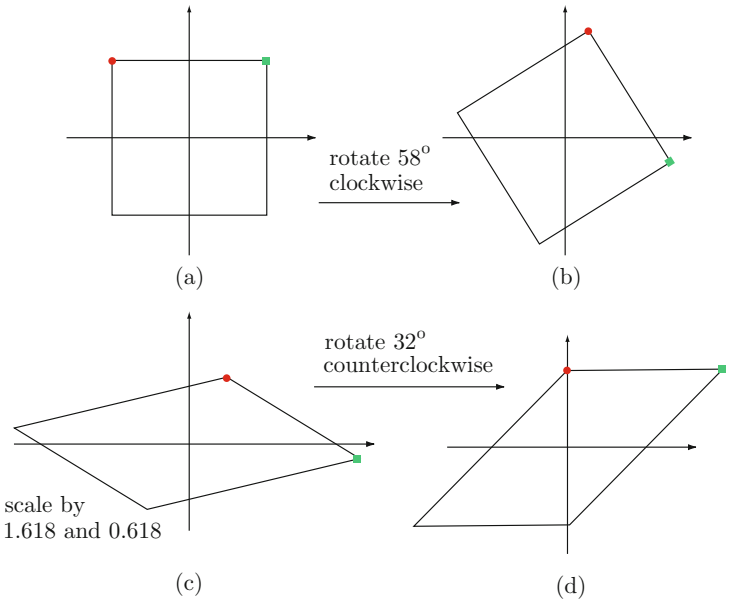


Figure 4.19: Shearing Decomposed into Rotation and Scaling.

4.2.10 Reconstructing Transformations

Given a sequence of two-dimensional transformations, we normally write the 3×3 matrix for each and then multiply the matrices. The result is another 3×3 matrix which is used to transform all the points of an object. An interesting question is: Given the points of an object before and after a transformation, can we reconstruct the transformation matrix from them?

The answer is yes! The general two-dimensional transformation matrix depends on six numbers, so all we need are six equations involving transformed points. Since each point consists of two numbers, three points are enough to reconstruct the transformation matrix. Given three points both before ($\mathbf{P}_1$, $\mathbf{P}_2$, $\mathbf{P}_3$) and after ($\mathbf{P}_1^*$, $\mathbf{P}_2^*$, $\mathbf{P}_3^*$) a transformation, we can write the three equations $\mathbf{P}_1^* = \mathbf{P}_1\mathbf{T}$, $\mathbf{P}_2^* = \mathbf{P}_2\mathbf{T}$, and $\mathbf{P}_3^* = \mathbf{P}_3\mathbf{T}$ and solve for the six elements of $\mathbf{T}$.

Example: The three points $(1,1)$, $(1,0)$, and $(0,1)$ are transformed to $(3,4)$, $(2,-1)$, and $(0,2)$, respectively. We write the general transformation $(x^*,y^*) = (ax+cy+m, bx+dy+n)$ for the three sets

$$\begin{aligned}(3,4) &= (a+c+m, b+d+n),\\ (2,-1) &= (a+m, b+n),\\ (0,2) &= (c+m, d+n),\end{aligned}$$

and this is easily solved to yield $a=3$, $b=2$, $c=1$, $d=5$, $m=-1$, and $n=-3$. The

transformation matrix is therefore

$$\mathbf{T} = \begin{pmatrix} 3 & 2 & 0 \\ 1 & 5 & 0 \\ -1 & -3 & 1 \end{pmatrix}.$$

◇ **Exercise 4.37:** Inverse transformations. From $\mathbf{P}^* = \mathbf{PT}$, we get $\mathbf{P}^*\mathbf{T}^{-1} = \mathbf{PTT}^{-1}$ or $\mathbf{P} = \mathbf{P}^*\mathbf{T}^{-1}$. We can therefore reconstruct an original point $\mathbf{P}$ from the transformed one, $\mathbf{P}^*$, if we know the inverse of the transformation matrix $\mathbf{T}$. In general, the inverse of the 3×3 matrix

$$\mathbf{T} = \begin{pmatrix} a & b & 0 \\ c & d & 0 \\ m & n & 1 \end{pmatrix}$$

is

$$\mathbf{T}^{-1} = \frac{1}{ad - bc} \begin{pmatrix} d & -b & 0 \\ -c & a & 0 \\ cn - dm & bm - an & 1 \end{pmatrix}. \tag{4.22}$$

Calculate the inverses of the transformation matrices for scaling, shearing, rotation, and translation, and discuss their properties.

◇ **Exercise 4.38:** Given that the four points

$$\mathbf{P}_1 = (0,0), \quad \mathbf{P}_2 = (0,1), \quad \mathbf{P}_3 = (1,1), \quad \text{and } \mathbf{P}_4 = (1,0)$$

are transformed to

$$\mathbf{P}_1^* = (0,0), \quad \mathbf{P}_2^* = (2,3), \quad \mathbf{P}_3^* = (8,4), \quad \text{and } \mathbf{P}_4^* = (6,1),$$

reconstruct the transformation matrix.

4.2.11 A Note

All the expressions derived so far for transformations are based on the basic relation $\mathbf{P}^* = \mathbf{PT}$. Some authors prefer the equivalent relation $\mathbf{P}^* = \mathbf{TP}$, which changes the mathematics somewhat. If we want the coordinates of the transformed point to be the same as before (i.e., $x^* = ax + cy + m$, $y^* = bx + dy + n$), we have to write the relation $\mathbf{P}^* = \mathbf{TP}$ in the form

$$\begin{pmatrix} x^* \\ y^* \\ 1 \end{pmatrix} = \begin{pmatrix} a & c & m \\ b & d & n \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}.$$

The first difference is that both $\mathbf{P}$ and $\mathbf{P}^*$ are columns instead of rows. This is because of the rules of matrix multiplication. The second difference is that the new transformation matrix $\mathbf{T}$ is the transpose of the original one. Hence, rotation, for example, is achieved by the matrices

$$\begin{pmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

for a clockwise rotation, and

$$\begin{pmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

for a counterclockwise rotation.

Similarly, translation is done by $\begin{pmatrix} 1 & 0 & m \\ 0 & 1 & n \\ 0 & 0 & 1 \end{pmatrix}$ instead of $\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ m & n & 1 \end{pmatrix}$.

4.2.12 Summary

The general two-dimensional affine transformation is given by $x^* = ax + cy + m$, $y^* = bx + dy + n$. This section lists the values or constraints that should be assigned to the four coefficients a, b, c, and d in order to obtain certain types of transformations (we ignore translations).

- A general affine transformation is obtained when $ad - bc \neq 0$. For $ad - bc = +1$, the transformation is rotation, and for $ad - bc = -1$, it is reflection.
- The case $ad - bc = 0$ corresponds to a singular transformation.
- The identity transformation is obtained for $a = d = 1$ and $b = c = 0$.
- An isometry is obtained by $a^2 + b^2 = c^2 + d^2 = 1$ and $ac + bd = 0$. An isometry is a transformation that preserves distances. If $\mathbf{P}$ and $\mathbf{Q}$ are two points on an object, then the distance d between them is preserved, meaning that the distance d between $\mathbf{P}^*$ and $\mathbf{Q}^*$ is the same. Rotations, reflections, and translations are isometries.
- A similarity is obtained for $a^2 + b^2 = c^2 + d^2$ and $ac + bd = 0$. A similarity is a transformation that preserves the ratios of lengths. A typical similarity is scaling, but it may be combined with rotation, reflection, and translation.
- An *equiareal* transformation (preserving areas) is obtained when $|ad - bc| = 1$.
- A shearing in the x direction is caused by $a = d = 1$ and $b = 0$. Similarly, a shearing in the y direction corresponds to $a = d = 1$ and $c = 0$.
- A uniform scaling is $a = d > 0$ and $b = c = 0$. (The identity is a special case of scaling.)
- A uniform reflection is $a = d < 0$ and $b = c = 0$.
- A rotation is the result of $a = d = \cos\theta$ and $b = -c = \sin\theta$.

4.3 Three-Dimensional Coordinate Systems

We now turn to transformations in three dimensions. In most cases, the mathematics of linear transformations is easy to extend from two dimensions to three dimensions, but the discussion in this section demonstrates that certain transformations, most notably rotations, are more complex in three dimensions because there are more directions about which to rotate and because the simple terms clockwise and counterclockwise no longer apply in three dimensions. We start with a short discussion of coordinate systems in three dimensions.

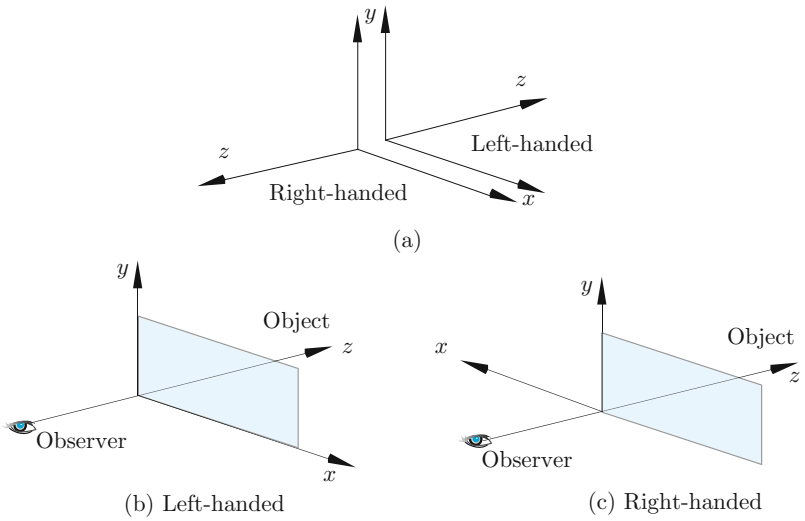


Figure 4.20: Three-Dimensional Coordinate Systems.

In two dimensions, there is only one Cartesian coordinate system, with two perpendicular axes labeled x and y (actually, the axes don't have to be perpendicular, but this is irrelevant for our discussion of transformations). A coordinate system in three dimensions consists similarly of three perpendicular axes labeled x, y, and z, but there are two such systems, a left-handed and a right-handed (Figure 4.20a), and they are different. A right-handed coordinate system is constructed by the following rule. Align your right thumb with the positive x axis and your right index finger with the positive y axis. Your right middle finger will then point in the direction of positive z. The rule for a left-handed system uses the left hand in a similar manner. It is also possible to define a left-handed coordinate system as the mirror image (reflection) of a right-handed system. Notice that one coordinate system cannot be transformed into the other by translating or rotating it.

The difference between left-handed and right-handed coordinate systems becomes important when a three-dimensional object is projected on a two-dimensional screen (Chapter 6). We assume that the screen is positioned at the xy plane with its origin (i.e., its bottom-left corner) at the origin of the three-dimensional system. We also assume that the object to be projected is located on the positive side of the z axis and the viewer is located on the negative side, looking at the projection of the image on the screen. Figure 4.20b shows that in a left-handed three-dimensional coordinate system, the directions of the positive x and y axes on the screen coincide with those of the three-dimensional x and y axes. However, in a right-handed system (Figure 4.20c) the two-dimensional x axis (on the screen) and the three-dimensional x axis point in opposite directions.

> Principle: Express co-ordinate ideas in similar form.
>
> This principle, that of parallel construction, requires that expressions of similar content and function should be outwardly similar. The likeness of form enables the reader to recognize more readily the likeness of content and function. Familiar instances from the Bible are the Ten Commandments, the Beatitudes, and the petitions of the Lord's Prayer.
>
> —W. Strunk Jr. and E. B. White, *The Elements of Style.*

4.4 Three-Dimensional Transformations

We derive three-dimensional transformations by extending the methods used in two-dimensional transformations, especially the concept of homogeneous coordinates. A three-dimensional point $\mathbf{P} = (x, y, z, 1)$ is transformed to a point $\mathbf{P}^* = (x^*, y^*, z^*, 1)$ by multiplying it by a 4×4 matrix

$$\mathbf{T} = \begin{pmatrix} a & b & c & p \\ d & e & f & q \\ h & i & j & r \\ l & m & n & s \end{pmatrix}. \tag{4.23}$$

The last column of $\mathbf{T}$ is not $(0, 0, 0, 1)^T$ and is used for projections. (See the discussion of n-point perspective on Page 319.) As a result, the product $\mathbf{PT}$ is the 4-tuple (X, Y, Z, H), where H equals $xp+yq+zr+s$ and is generally not 1. The three coordinates (x^*, y^*, z^*) of $\mathbf{P}^*$ are obtained by dividing (X, Y, Z) by H. Hence, $(x^*, y^*, z^*) = (X/H, Y/H, Z/H)$.

The top left 3×3 part of $\mathbf{T}$ is responsible for scaling and reflection (a, e, and j), shearing (b, c, f and d, h, i), and rotation (all nine elements). The three quantities l, m, and n are responsible for translation, and the only new parameters are those in the last column (p, q, r, s).

To understand the meaning of s, we examine the matrix $\mathbf{T} = \begin{pmatrix} 1 & & & \\ & 1 & & \\ & & 1 & \\ & & & s \end{pmatrix}$. Multiplying $\mathbf{P}$ by $\mathbf{T}$ transforms $(x, y, z, 1)$ into (x, y, z, s), so the new point has coordinates

$(x/s, y/s, z/s)$. The parameter s is therefore responsible for global scaling (by a factor of $1/s$). Its effect is identical to transforming by $\begin{pmatrix} 1/s & & \\ & 1/s & \\ & & 1/s \\ & & & 1 \end{pmatrix}$.

Translation in three dimensions is a direct extension of the two-dimensional case. A point can be translated in the direction of any of the coordinate axes.

Scaling in three dimensions is simple. An object can be scaled about the origin along any of the three coordinate axes. To scale about another point $\mathbf{P}_0$, a sequence of three transformations is needed. The point should be translated to the origin, the scaling performed, and the point translated back. Notice that scaling an object is done by scaling all its points. Scaling a point does not change its dimensions (a point has no dimensions) but simply moves it to another location.

Shearing in three dimensions is difficult to visualize. It is controlled by the six off-diagonal matrix elements b, c, f, d, h, and i, which is why many variations are possible. Perhaps the best way to become familiar with three-dimensional shearing is to experiment with the effect of varying each of the six parameters. Figure 4.21 shows a few possible shearings of a rectangular box.

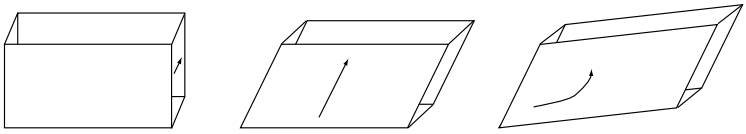


Figure 4.21: Shearing in Three Dimensions.

> Shearing: A transformation in which all points along a given line L remain fixed while other points are shifted parallel to L by a distance proportional to their perpendicular distance from L. Shearing a plane figure does not change its area. This can also be generalized to three dimensions, where planes are translated instead of lines.
>
> —Eric W. Weisstein, http://mathworld.wolfram.com/Shear.html

4.4.1 Reflection

It is easy to reflect a point (x, y, z) about any of the three coordinate planes xy, xz, or yz. All that is needed is to change the sign of one of the point's coordinates. In this section, we discuss and explain the general case where an arbitrary plane and a point are given and we want to reflect the point about the plane. We proceed in three steps as follows: (1) We discuss planes and their equations (there is a similar discussion in Section 9.2.2), (2) we show how to determine the distance of a point from a given plane, and (3) we explain how to compute the reflection of a point about a plane.

The (implicit) equation of a straight line is $Ax+By+C = 0$, where A or B but not both can be zero. The equation of a flat plane is the direct extension $Ax+By+Cz+D = 0$, where A, B, and C cannot all be zero. Four equations are needed to calculate the four unknown coefficients A, B, C, and D. On the other hand, we know that any three independent (i.e., noncollinear) points $\mathbf{P}_i = (x_i, y_i, z_i)$, $i = 1, 2, 3$ define a plane. Thus, we can write a set of four equations, three of which are based on three given points and

the fourth one expressing the condition that a general point (x, y, z) lies on the plane

$$
\begin{aligned}
0 &= \begin{vmatrix} x & y & z & 1 \\ x_1 & y_1 & z_1 & 1 \\ x_2 & y_2 & z_2 & 1 \\ x_3 & y_3 & z_3 & 1 \end{vmatrix} \\
&= x \begin{vmatrix} y_1 & z_1 & 1 \\ y_2 & z_2 & 1 \\ y_3 & z_3 & 1 \end{vmatrix} - y \begin{vmatrix} x_1 & z_1 & 1 \\ x_2 & z_2 & 1 \\ x_3 & z_3 & 1 \end{vmatrix} + z \begin{vmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ x_3 & y_3 & 1 \end{vmatrix} - \begin{vmatrix} x_1 & y_1 & z_1 \\ x_2 & y_2 & z_2 \\ x_3 & y_3 & z_3 \end{vmatrix}.
\end{aligned}
$$

We cannot solve this system of equations because x, y, and z can have any values, but we don't need to solve it! We just have to guarantee that this system has a solution. In general, a system of linear algebraic equations has a solution if and only if its determinant is zero. The expression below assumes this and also expands the determinant by its top row:

$$
\begin{aligned}
0 &= \begin{vmatrix} x & y & z & 1 \\ x_1 & y_1 & z_1 & 1 \\ x_2 & y_2 & z_2 & 1 \\ x_3 & y_3 & z_3 & 1 \end{vmatrix} \\
&= x \begin{vmatrix} y_1 & z_1 & 1 \\ y_2 & z_2 & 1 \\ y_3 & z_3 & 1 \end{vmatrix} - y \begin{vmatrix} x_1 & z_1 & 1 \\ x_2 & z_2 & 1 \\ x_3 & z_3 & 1 \end{vmatrix} + z \begin{vmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ x_3 & y_3 & 1 \end{vmatrix} - \begin{vmatrix} x_1 & y_1 & z_1 \\ x_2 & y_2 & z_2 \\ x_3 & y_3 & z_3 \end{vmatrix}.
\end{aligned}
$$

This is of the form $Ax + By + Cz + D = 0$, so we conclude that

$$
A = \begin{vmatrix} y_1 & z_1 & 1 \\ y_2 & z_2 & 1 \\ y_3 & z_3 & 1 \end{vmatrix} \quad B = - \begin{vmatrix} x_1 & z_1 & 1 \\ x_2 & z_2 & 1 \\ x_3 & z_3 & 1 \end{vmatrix} \quad C = \begin{vmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ x_3 & y_3 & 1 \end{vmatrix} \quad D = - \begin{vmatrix} x_1 & y_1 & z_1 \\ x_2 & y_2 & z_2 \\ x_3 & y_3 & z_3 \end{vmatrix}. \tag{4.24}
$$

◇ **Exercise 4.39:** Derive the expression of the plane containing the z axis and passing through the point $(1, 1, 0)$.

◇ **Exercise 4.40:** In the plane equation $Ax + By + Cz + D = 0$ if $D = 0$, then the plane passes through the origin. Assuming $D \neq 0$, we can write the same equation as $x/a + y/b + z/c = 1$, where $a = -D/A$, $b = -D/B$, and $c = -D/C$. What is the geometrical interpretation of a, b, and c?

> We operate with nothing but things which do not exist, with lines, planes, bodies, atoms, divisible time, divisible space—how should explanation even be possible when we first make everything into an image, into our own image!
>
> —Friedrich Nietzsche.

In some practical applications, the normal to the plane and one point on the plane are known. It is easy to derive the plane equation in such a case.

We assume that $\mathbf{N}$ is the (known) normal vector to the plane, $\mathbf{P}_1$ is a known point on the plane, and $\mathbf{P}$ is an arbitrary point in the plane. The vector $\mathbf{P}-\mathbf{P}_1$ is perpendicular

to $\mathbf{N}$, so their dot product $\mathbf{N} \bullet (\mathbf{P} - \mathbf{P}_1)$ equals zero. Since the dot product is associative, we can write $\mathbf{N} \bullet \mathbf{P} = \mathbf{N} \bullet \mathbf{P}_1$. The dot product $\mathbf{N} \bullet \mathbf{P}_1$ is just a number, to be denoted by s, so we get

$$\mathbf{N} \bullet \mathbf{P} = s \quad \text{or} \quad N_x x + N_y y + N_z z - s = 0. \tag{4.25}$$

Equation (4.25) can now be written as $Ax + By + Cz + D = 0$, where $A = N_x$, $B = N_y$, $C = N_z$, and $D = -s = -\mathbf{N} \bullet \mathbf{P}_1$. The three unknowns A, B, and C are the components of the normal vector, and D can be calculated from any known point $\mathbf{P}_1$ on the plane. The expression $\mathbf{N} \bullet \mathbf{P} = s$ is a useful equation of the plane and is used in many applications.

◇ **Exercise 4.41:** Given $\mathbf{N}(u, w) = (1, 1, 1)$ and $\mathbf{P}_1 = (1, 1, 1)$, calculate the plane equation.

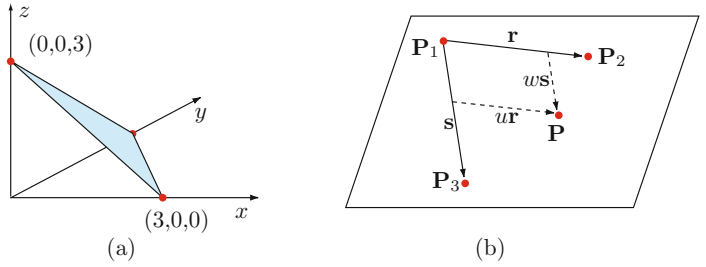


Figure 4.22: (a). A Plane. (b) Three Points on a Plane.

Note that the direction in which the normal is pointing is irrelevant for the plane equation. Substituting $(-A, -B, -C)$ for (A, B, C) would also change the sign of D, resulting in the same equation. However, the direction of the normal is important when a surface is to be shaded. We want the normal, in such a case, to point *outside* the surface. Often, this has to be done manually since the computer has no concept of the shape of the object in question and the meaning of the terms "inside" and "outside." However, in cases where a plane is defined by three points, the direction of the normal can be specified by arranging the three points (in the data structure in memory) in a certain order.

It is also easy to derive the equation of a plane when three points on the plane, $\mathbf{P}_1$, $\mathbf{P}_2$, and $\mathbf{P}_3$, are known. In order for the points to define a plane, they should not be collinear. We consider the vectors $\mathbf{r} = \mathbf{P}_2 - \mathbf{P}_1$ and $\mathbf{s} = \mathbf{P}_3 - \mathbf{P}_1$ a local coordinate system on the plane. Any point $\mathbf{P}$ on the plane can be expressed as a linear combination $\mathbf{P} = u\mathbf{r} + w\mathbf{s}$, where u and w are real numbers. Since $\mathbf{r}$ and $\mathbf{s}$ are local coordinates on the plane, the position of point $\mathbf{P}$ relative to the origin is expressed as (Figure 4.22b)

$$\mathbf{P}(u, w) = \mathbf{P}_1 + u\mathbf{r} + w\mathbf{s}, \quad -\infty < u, w < \infty. \tag{4.26}$$

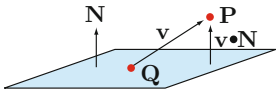


Figure 4.23: Distance of a Point from a Plane.

◇ **Exercise 4.42:** Given the three points $\mathbf{P}_1 = (3, 0, 0)$, $\mathbf{P}_2 = (0, 3, 0)$, and $\mathbf{P}_3 = (0, 0, 3)$, write the equation of the plane defined by them.

The next step is to determine the distance between a point and a plane. Given the point $\mathbf{P} = (x, y, z)$ and the plane $Ax + By + Cz + D = 0$, we select an arbitrary point $\mathbf{Q} = (x_0, y_0, z_0)$ on the plane. Since $\mathbf{Q}$ is on the plane, it satisfies $Ax_0+By_0+Cz_0+D = 0$ or $-Ax_0 - By_0 - Cz_0 = D$. We construct the vector $\mathbf{v}$ from $\mathbf{Q}$ to $\mathbf{P}$ as the difference $\mathbf{v} = \mathbf{P} - \mathbf{Q} = (x - x_0, y - y_0, z - z_0)$. Figure 4.23 shows that the required distance (the size of the vector from the plane to $\mathbf{P}$ that's perpendicular to the plane) is the component $\mathbf{v}_N$ of $\mathbf{v}$ in the direction of the normal $\mathbf{N} = (A, B, C)$. This component is given by

$$\begin{aligned}\mathbf{v}_N &= \frac{|\mathbf{v} \bullet \mathbf{N}|}{|\mathbf{N}|} = \frac{|A(x - x_0) + B(y - y_0) + C(z - z_0)|}{\sqrt{A^2 + B^2 + C^2}} \\ &= \frac{|Ax + By + Cz - Ax_0 - By_0 - Cz_0|}{\sqrt{A^2 + B^2 + C^2}} \\ &= \frac{|Ax + By + Cz + D|}{\sqrt{A^2 + B^2 + C^2}}.\end{aligned} \tag{4.27}$$

If we omit the absolute value, then the distance becomes a signed quantity. We can think of the plane as if it divides all of space into two parts, one in the direction of $\mathbf{N}$ and the other on the other side of the plane. The distance is positive if $\mathbf{P}$ is located in that part of space pointed to by the normal (which is the case in Figure 4.23), and it is negative in the opposite case.

◇ **Exercise 4.43:** What's the distance of a plane from the origin?

Now that we can figure out the distance between a point and a plane, the last step is to reflect a point about a given plane. We start with a point $\mathbf{P} = (x, y, z)$ and a plane $Ax + By + Cz + D = 0$. We denote the normal unit vector by $\mathbf{N} = (A, B, C)/\sqrt{A^2 + B^2 + C^2}$ and the (signed) distance between $\mathbf{P}$ and the plane by d. To get from $\mathbf{P}$ to the plane, we have to travel a distance d in the direction of $\mathbf{N}$. To arrive at the reflection point $\mathbf{P}^*$, we should travel another d units in the same direction. Thus, the reflection $\mathbf{P}^*$ of $\mathbf{P}$ is given by

$$\mathbf{P}^* = \mathbf{P} - 2d\mathbf{N} = \mathbf{P} - \frac{2(Ax + By + Cz + D)}{A^2 + B^2 + C^2}(A, B, C). \tag{4.28}$$

◇ **Exercise 4.44:** Why $\mathbf{P} - 2d\mathbf{N}$ and not $\mathbf{P} + 2d\mathbf{N}$?

> Most neurotics have been mindful of their five W's since grammar school: why, why, why, why, why.
>
> —Terri Guillemets.

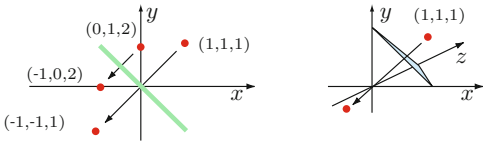


Figure 4.24: Reflection in Three Dimensions: Examples.

Examples: We select (Figure 4.24a) the plane $x+y=0$ and the point $\mathbf{P} = (1,1,1)$. Equation (4.28) becomes

$$\mathbf{P}^* = (1,1,1) - \frac{2(1+1)}{1+1+0}(1,1,0) = (-1,-1,1).$$

Similarly, point $\mathbf{P} = (0,1,2)$ is reflected to

$$\mathbf{P}^* = (0,1,2) - \frac{2(0+1)}{1+1+0}(1,1,0) = (-1,0,2).$$

We now select (Figure 4.24b) the plane $x+y+z-1=0$ and the point $\mathbf{P} = (1,1,1)$. Equation (4.28) becomes

$$\mathbf{P}^* = (1,1,1) - \frac{2(1+1+1-1)}{1+1+1}(1,1,1) = -\frac{1}{3}(1,1,1).$$

Similarly, point $\mathbf{P} = (0,0,0)$ is reflected to

$$\mathbf{P}^* = (0,0,0) - \frac{2(0+0+0-1)}{1+1+1}(1,1,1) = \frac{2}{3}(1,1,1).$$

The special case of a reflection about one of the coordinate planes is also obtained from Equation (4.28). The equation of the xy plane, for example, is $z=0$, where Equation (4.28) yields

$$\mathbf{P}^* = (x,y,z) - \frac{2(0+0+z+0)}{0^2+0^2+1^2}(0,0,1) = (x,y,-z).$$

4.4.2 Rotation

Rotation in three dimensions is difficult to visualize and is often confusing. One approach to rotations is to write three rotation matrices that rotate about the three coordinate axes:

$$\begin{pmatrix} \cos\theta & -\sin\theta & 0 & 0 \\ \sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \begin{pmatrix} \cos\theta & 0 & -\sin\theta & 0 \\ 0 & 1 & 0 & 0 \\ \sin\theta & 0 & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta & 0 \\ 0 & \sin\theta & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}. \tag{4.29}$$

Let's look at the first of these matrices. Its third row and third column are $(0, 0, 1, 0)$, which is why multiplying a point $(x, y, z, 1)$ by this matrix leaves its z coordinate unchanged. The sines and cosines in the first two rows and two columns mix up the x and y coordinates in a way similar to a two-dimensional rotation, Equation (4.4). Thus, this transformation matrix causes a rotation about the z axis. The two other matrices rotate about the y and x axes.

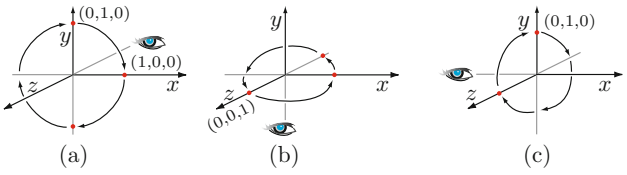


Figure 4.25: Rotating About the Coordinate Axes.

> Okay, so I assume going into this tutorial that you know how to perform matrix multiplication. I don't care to explain it, and it's available all over the Internet. However, once you know how to perform that operation, you should be good to go for this tutorial.
>
> (Found on the Internet).

It is therefore easy to identify the axis of rotation for each of the three rotation matrices of Equation (4.29), but what about their direction of rotation? To figure out the directions, we select $\theta = 90°$ and substitute $\sin\theta = 1$ and $\cos\theta = 0$. Simple tests in a right-handed coordinate system show that the first matrix of Equation (4.29) (rotation about the z axis) rotates point $(1, 0, 0)$ to $(0, -1, 0)$ and point $(0, 1, 0)$ to $(1, 0, 0)$. Thus, when we observe this 90° rotation looking in the direction of positive z, the rotation is counterclockwise (Figure 4.25a). The second matrix, however, behaves differently. It rotates point $(1, 0, 0)$ to $(0, 0, -1)$ and point $(0, 0, 1)$ to $(1, 0, 0)$. When we observe this 90° rotation about the y axis looking in the direction of positive y, the rotation is clockwise (Figure 4.25b). The third matrix (rotation about the x axis) rotates point $(0, 1, 0)$ to $(0, 0, -1)$ and point $(0, 0, 1)$ to $(0, 1, 0)$. When we observe this 90° rotation looking in the direction of positive x, the rotation is counterclockwise (Figure 4.25c).

We therefore decide (somewhat arbitrarily) to switch the signs (positive and negative) of the sine functions in the matrices that rotate about the z and x axes. The

result,

$$\begin{pmatrix} \cos\theta & \sin\theta & 0 & 0 \\ -\sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \begin{pmatrix} \cos\theta & 0 & -\sin\theta & 0 \\ 0 & 1 & 0 & 0 \\ \sin\theta & 0 & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\theta & \sin\theta & 0 \\ 0 & -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad (4.30)$$

is a set of three rotation matrices that rotate a point about the three coordinate axes in such a way that if we look in the positive direction of that axis, the rotation is clockwise.

(Surprisingly, it turns out that there is an elegant way to specify the direction of rotation that's generated by the rotation matrices of Equation (4.29), and this is described below.)

The rotation matrices of Equations (4.29) and (4.30) are simple but not very useful because in practice we rarely know how to break a general rotation into three rotations about the coordinate axes. There are some cases, however, where rotations about the coordinate axes are common. One such case is discussed in Section 5.2; two more are presented here.

Case 1: Rotations about the coordinate axes are common in the motion of a submarine or an airplane. These vehicles have three degrees of freedom and have three natural, mutually perpendicular axes of rotation that are called *roll*, *pitch*, and *yaw* (Figure 4.26). Roll is a rotation about the direction of motion of the vehicle. An airplane rolls when it banks by dipping one wing and lifting the other. Pitch is an up or down rotation about an axis that goes through the wings. An airplane uses its elevators for this. Yaw is a left–right rotation about a vertical axis, accomplished by the rudder. These terms originated with sailors because a ship can yaw and also has limited roll and pitch capabilities.

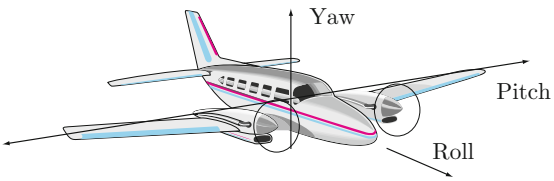


Figure 4.26: Roll, Pitch, and Yaw.

Case 2: Another example of an application where rotations about the three coordinate axes are common is L-systems. This is a system of formal notation developed by the biologist Aristid Lindenmayer (hence the "L") in 1968 as a tool to describe the morphology of plants [Lindenmayer 68]. In the 1970s, this notation was adopted by computer scientists and used to define formal languages. Since 1984, it has also been used to describe and draw many types of fractals. Today, L-systems are used to generate tilings, geometric art, and even music.

The main idea of L-systems is to specify a complex object by (1) defining an initial simple object, called the *axiom*, and (2) writing rules that show how to replace parts

of the axiom. The rules are written in terms of *turtle moves*, a concept originally introduced in the LOGO programming language [Abelson and diSessa 82]. L-systems, however, specify the structure of three-dimensional objects, so the turtle must move in three dimensions and can rotate about its three main axes. For more information on L-systems, see [Prusinkiewicz 89].

It has already been mentioned that rotation in three dimensions is more complex than in two dimensions. One reason for this is that rotation in two dimensions is about a point, whereas rotation in three dimensions is about an axis (any axis, not just one of the three coordinate axes). Another reason is that the direction of rotation in two dimensions can be only clockwise or counterclockwise, but the direction of rotation in three dimensions is more complex to specify. The rotation is about an axis, but its direction, clockwise or counterclockwise, about this axis depends on how we look at the axis. Thus, a general rule is needed to specify the direction of a three-dimensional rotation unambiguously. We state such a rule for the rotation matrices of Equation (4.29).

The direction of a three-dimensional rotation generated by the matrices of (4.29) in a right-handed coordinate system is determined by the following rule: Write down the sequence "x, y, z" and erase the symbol that corresponds to the axis of rotation. The two remaining symbols are denoted by l and r. Draw the coordinate axes such that the positive direction of l will be up and the positive direction of r will be to the right. (This is not a necessary requirement, but it conforms to Figure 4.27.) The rotation will then be from positive r to positive l to negative r to negative l (Figure 4.27 and see also Exercise 6.13).

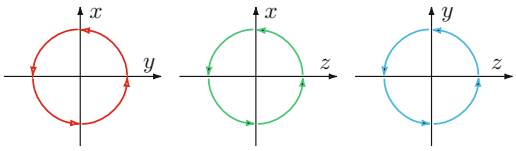


Figure 4.27: Direction of Three-Dimensional Rotations.

Example: A rotation about the z axis produced by the leftmost matrix of (4.29). After erasing z, the two symbols left are x and y. We draw the coordinate axes such that positive x is up and positive y is to the right. The matrix produces counterclockwise rotation. To achieve clockwise rotation, either use a negative angle or the inverse of the rotation matrix. Inverting our rotation matrices is especially easy and requires only that we change the signs of the sine functions.

Example: Consider the following compound transformation: (1) a translation by l, m, and n units along the three coordinate axes, (2) a rotation of θ degrees about the x axis, (3) a rotation of ϕ degrees about the y axis, and (4) the reverse translation. The

four transformation matrices are

$$\mathbf{T}_r = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ l & m & n & 1 \end{pmatrix}, \quad \mathbf{T}_{rr} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -l & -m & -n & 1 \end{pmatrix},$$

$$\mathbf{R}_x = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\theta & \sin\theta & 0 \\ 0 & -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad \mathbf{R}_y = \begin{pmatrix} \cos\phi & 0 & -\sin\phi & 0 \\ 0 & 1 & 0 & 0 \\ \sin\phi & 0 & \cos\phi & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

Their product equals the 4×4 matrix

$$\begin{aligned} \mathbf{T} &= \mathbf{T}_r\mathbf{R}_x\mathbf{R}_y\mathbf{T}_{rr} \\ &= \begin{pmatrix} \cos\phi & 0 & -\sin\phi & 0 \\ \sin\phi\sin\theta & \cos\theta & \cos\phi\sin\theta & 0 \\ \cos\theta\sin\phi & -\sin\theta & \cos\phi\cos\theta & 0 \\ -l + l\cos\phi & -m & [-2n + n\cos(\phi-\theta) & 1 \\ \quad +m\cos(\phi-\theta)/2 & \quad +m\cos\theta & \quad +n\cos(\phi+\theta) & \\ \quad -m\cos(\phi+\theta)/2 & \quad -n\sin\theta & \quad -2l\sin\phi & \\ \quad +n\sin(\phi-\theta)/2 & & \quad -m\sin(\phi-\theta) & \\ \quad +n\sin(\phi+\theta)/2 & & \quad +m\sin(\phi+\theta)]/2 & \end{pmatrix}. \end{aligned}$$

Substituting the values $\theta = 30^\circ$, $\phi = 45^\circ$, and $l = m = n = -1$, we get the 4×4 matrix

$$\mathbf{T} = \begin{pmatrix} 0.7071 & 0 & -0.7071 & 0 \\ 0.3540 & 0.866 & 0.3540 & 0 \\ 0.6124 & -0.50 & 0.6124 & 0 \\ -0.673 & 0.634 & 0.7410 & 1 \end{pmatrix}.$$

A point at $(1, 2, 3)$, for example, is transformed by $\mathbf{T}$ to the point

$$(1, 2, 3, 1)\mathbf{T} = (2.5793, 0.866, 2.5791, 1).$$

◇ **Exercise 4.45:** Do the same operations for the compound transformation $\mathbf{T}_r\mathbf{R}_x\mathbf{T}_{rr}$.

4.4.3 General Rotations

In practice, we generally don't know how to express an arbitrary rotation as a product of rotations about the coordinate axes, so we have to derive the important transformation of general rotation explicitly. The problem is easy to state. A point $\mathbf{P}$ is to be rotated through an angle θ about a specified axis. It is important to realize that there is a difference between an axis and a vector. A vector is fully specified by three numbers. It has direction and magnitude, but no specific location in space. An axis has both direction and location (it starts at a certain point), but its magnitude is normally irrelevant. A full specification of an axis requires a start point and a vector, a total of six numbers.

(However, because the magnitude of the vector is irrelevant, it can be represented by two numbers only.) In order to simplify our derivation, we assume that our axis of rotation starts at the origin. If it starts at point $\mathbf{P}_0$, we have to precede the rotation by a translation of $\mathbf{P}_0$ to the origin and follow the rotation by the inverse translation (see also Section 24.3.10 for a discussion of rotations in connection with the discrete cosine transform (DCT)).

We therefore denote by $\mathbf{u}$ a unit vector located on an axis that starts at the origin. We can now fully specify a general rotation in three dimensions by four numbers—the rotation angle θ and the three components of $\mathbf{u}$. The rotated point $\mathbf{P}$ ends up at $\mathbf{P}^*$. We connect $\mathbf{P}$ to the origin and call the resulting vector $\mathbf{r}$. Rotating point $\mathbf{P}$ to $\mathbf{P}^*$ is identical to rotating vector $\mathbf{r}$ to $\mathbf{r}^*$.

Figure 4.28a shows that the component OC of $\mathbf{r}$ along $\mathbf{u}$ is left unchanged, but the component CP is rotated to CP*. The distance OC is seen from the diagram to be $(\mathbf{r}\bullet\mathbf{u})$, so the vector $\vec{\mathrm{OC}}$ can be written $(\mathbf{r}\bullet\mathbf{u})\mathbf{u}$. From $\mathbf{r}=\vec{\mathrm{OC}}+\vec{\mathrm{CP}}$, we get $\vec{\mathrm{CP}}=\mathbf{r}-(\mathbf{r}\bullet\mathbf{u})\mathbf{u}$ or, in terms of magnitudes, $|\vec{\mathrm{CP}}|=|\mathbf{r}-(\mathbf{r}\bullet\mathbf{u})\mathbf{u}|$. It can also be seen from the diagram that $|\vec{\mathrm{CP}}|=|\mathbf{r}|\sin\phi$. Since $\mathbf{u}$ is a unit vector, we can write $|\mathbf{u}\times\mathbf{r}|=|\mathbf{r}|\sin\phi$. We thus obtain $|\vec{\mathrm{CP}}|=|\mathbf{r}-(\mathbf{r}\bullet\mathbf{u})\mathbf{u}|=|\mathbf{u}\times\mathbf{r}|$.

Figure 4.28b shows the situation when looking from the origin in the positive $\mathbf{u}$ direction. (The diagram shows the tail of $\mathbf{u}$.) Note that the vector $\vec{\mathrm{CQ}}$ is perpendicular to both $\mathbf{u}$ and $\mathbf{r}$, so it is in the direction of $\mathbf{u}\times\mathbf{r}$.

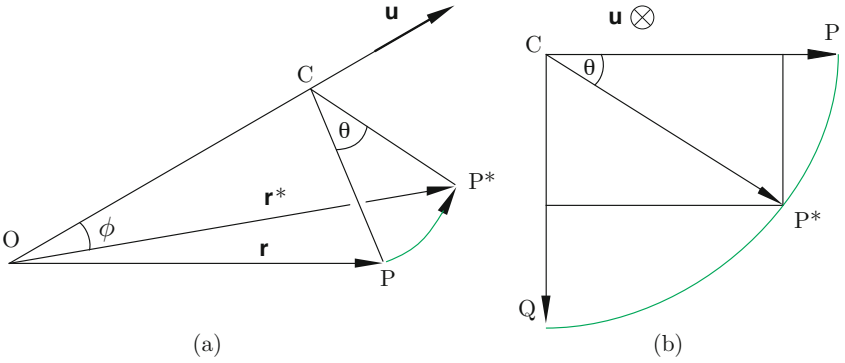


Figure 4.28: A General Rotation.

The next step is to resolve CP* into its components. From Figure 4.28b, we get

$$\vec{\mathrm{CP}}^*=\cos\theta[\mathbf{r}-(\mathbf{r}\bullet\mathbf{u})\mathbf{u}]+\sin\theta[\mathbf{r}-(\mathbf{r}\bullet\mathbf{u})\mathbf{u}]=\cos\theta[\mathbf{r}-(\mathbf{r}\bullet\mathbf{u})\mathbf{u}]+\sin\theta(\mathbf{u}\times\mathbf{r}),$$

which can be used to express $\mathbf{r}^*$:

$$\mathbf{r}^*=\vec{\mathrm{OC}}+\vec{\mathrm{CP}}^*=(\mathbf{r}\bullet\mathbf{u})\mathbf{u}+\cos\theta[\mathbf{r}-(\mathbf{r}\bullet\mathbf{u})\mathbf{u}]+\sin\theta(\mathbf{u}\times\mathbf{r}). \tag{4.31}$$

Using Equations (A.3) and (A.5) (Page 1290), we can rewrite this as $\mathbf{r}^* = (\mathbf{u}\mathbf{u}^T)\mathbf{r} + \cos\theta\mathbf{r} - \cos\theta(\mathbf{u}\mathbf{u}^T)\mathbf{r} + \sin\theta\mathbf{U}\mathbf{r}$, where

$$\mathbf{U} = \begin{pmatrix} 0 & -u_z & u_y \\ u_z & 0 & -u_x \\ -u_y & u_x & 0 \end{pmatrix}.$$

The result can now be summarized as $\mathbf{r}^* = \mathbf{M}\mathbf{r}$, where

$$\begin{aligned} \mathbf{M} &= \mathbf{u}\mathbf{u}^T + \cos\theta(\mathbf{I} - \mathbf{u}\mathbf{u}^T) + \sin\theta\mathbf{U} \qquad (4.32) \\ &= \begin{bmatrix} u_x^2 + \cos\theta(1-u_x^2) & u_xu_y(1-\cos\theta) - u_z\sin\theta & u_xu_z(1-\cos\theta) + u_y\sin\theta \\ u_xu_y(1-\cos\theta) + u_z\sin\theta & u_y^2 + \cos\theta(1-u_y^2) & u_yu_z(1-\cos\theta) - u_x\sin\theta \\ u_xu_z(1-\cos\theta) - u_y\sin\theta & u_yu_z(1-\cos\theta) + u_x\sin\theta & u_z^2 + \cos\theta(1-u_z^2) \end{bmatrix}. \end{aligned}$$

Direction cosines. If $\mathbf{v} = (v_x, v_y, v_z)$ is a three-dimensional vector, its *direction cosines* are defined as

$$N_1 = \frac{v_x}{|\mathbf{v}|}, \quad N_2 = \frac{v_y}{|\mathbf{v}|}, \quad N_3 = \frac{v_z}{|\mathbf{v}|}.$$

These are the cosines of the angles between the direction of $\mathbf{v}$ and the three coordinate axes. It is easy to verify that $N_1^2 + N_2^2 + N_3^2 = 1$. If $\mathbf{u} = (u_x, u_y, u_z)$ is a unit vector, then $|\mathbf{u}| = 1$ and u_x, u_y, and u_z are the direction cosines of $\mathbf{u}$.

It can be shown that a rotation through an angle $-\theta$ is performed by the transpose $\mathbf{M}^T$. Consider the two successive and opposite rotations $\mathbf{r}^* = \mathbf{M}\mathbf{r}$ and $\mathbf{r}' = \mathbf{M}^T\mathbf{r}^*$. On the one hand, they can be expressed as the product $\mathbf{r}' = \mathbf{M}^T\mathbf{r}^* = \mathbf{M}^T\mathbf{M}\mathbf{r}$. On the other hand, they rotate in opposite directions, so they return all points to their original positions; therefore $\mathbf{r}'$ must be equal to $\mathbf{r}$. We end up with $\mathbf{r} = \mathbf{M}^T\mathbf{M}\mathbf{r}$ or $\mathbf{M}\mathbf{M}^T = \mathbf{I}$, where $\mathbf{I}$ is the identity matrix. The transpose $\mathbf{M}^T$ therefore equals the inverse, $\mathbf{M}^{-1}$, of $\mathbf{M}$, which shows that a rotation matrix $\mathbf{M}$ is orthogonal.

Example: Consider a rotation about the z axis. The rotation axis is $\mathbf{u} = (0, 0, 1)$, resulting in

$$\mathbf{u}\mathbf{u}^T = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \text{ and } \mathbf{U} = \begin{pmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \text{ and hence } \mathbf{M} = \begin{pmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{pmatrix},$$

which is the familiar rotation matrix about the z axis. It is identical to the z-rotation matrix of Equation (4.29), so we conclude that it rotates counterclockwise when viewed from the direction of positive z.

The general rotation matrix of Equation (4.32) can also be constructed as the product of five simple rotations about various coordinate axes. Given a unit vector $\mathbf{u} = (u_x, u_y, u_z)$, consider the following rotations.

1. Rotate $\mathbf{u}$ about the z axis into the xz plane, so its y coordinate becomes zero. This is done by a rotation matrix of the form

$$\mathbf{A} = \begin{bmatrix} \cos\psi & -\sin\psi & 0 \\ \sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix},$$

and the angle ψ of rotation can be computed from the requirement that the y component of vector $\mathbf{v} = \mathbf{uA}$ be zero. This component is $-u_x \sin\psi + u_y \cos\psi$, which implies $\cos\psi = u_x/\sqrt{u_x^2 + u_y^2}$ and $\sin\psi = u_y/\sqrt{u_x^2 + u_y^2}$. Notice that rotating $\mathbf{u}$ does not affect its magnitude, so $\mathbf{v}$ is also a unit vector. In addition, since the rotation is about the z axis, the z component of $\mathbf{u}$ does not change, so $v_z = u_z$.

2. Rotate vector $\mathbf{v}$ about the y axis until it coincides with the z axis. This is accomplished by the matrix

$$\mathbf{B} = \begin{bmatrix} \cos\phi & 0 & \sin\phi \\ 0 & 1 & 0 \\ -\sin\phi & 0 & \cos\phi \end{bmatrix}.$$

The angle ϕ of rotation is computed from the dot product $\cos\phi = \mathbf{v} \cdot (0, 0, 1) = v_z = u_z$, implying that $\sin\phi = \sqrt{1 - u_z^2}$. Since $\mathbf{v}$ is a unit vector, it is rotated by $\mathbf{B}$ to vector $(0, 0, 1)$.

3. Rotate $(0, 0, 1)$ about the z axis through an angle θ. This is done by matrix

$$\mathbf{C} = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

This is a trivial rotation that does not change $(0, 0, 1)$.

4. Rotate the result of step 3 by $\mathbf{B}^{-1}$ (which equals $\mathbf{B}^T$).
5. Rotate the result of step 4 by $\mathbf{A}^{-1}$ (which equals $\mathbf{A}^T$).

When these five steps are performed on a point (x, y, z), the effect is to rotate the point through an angle θ about $\mathbf{u}$. In practice, the five steps are combined by multiplying the five matrices above, as shown in the listing of Figure 4.29. The result is identical to Equation (4.32).

```
tm=Sqrt[x^2+y^2];
a={{x/tm,-y/tm,0},{y/tm,x/tm,0},{0,0,1}};
b={{z,0,Sqrt[1-z^2]},{0,1,0},{-Sqrt[1-z^2],0,z}};
c={{Cos[t],-Sin[t],0},{Sin[t],Cos[t],0},{0,0,1}};
FullSimplify[a.b.c.Transpose[b].Transpose[a] /. x^2+y^2->1-z^2]
```

Figure 4.29: *Mathematica* Code for a General Rotation.

4.4.4 Givens Rotations

The general rotation matrix, Equation (4.32), can be constructed for any general rotation in three dimensions. Given such a matrix $\mathbf{A}$, it is possible to reduce it to a product of rotation matrices that cause the same rotation by performing a sequence of rotations about the coordinate axes. This process, first described in [Givens 58], is based on the QR decomposition of matrices, a subject discussed in any text on matrices (and also in Section 24.3.8), and it results in a set of *Givens rotations*. Each Givens rotation matrix

$\mathbf{T}_{i,j}$ is identified by two indexes, i and j, where $i > j$. The matrix is an identity matrix except for the two diagonal elements (i,i) and (j,j) that are cosines of some angle and for the two off-diagonal elements (i,j) and (j,i) that are the $\pm\sin$ of the same angle. Specifically, $\mathbf{T}_{i,j}[i,i] = \mathbf{T}_{i,j}[j,j] = c$ and $\mathbf{T}_{i,j}[j,i] = -\mathbf{T}_{i,j}[i,j] = s$, where $c = \mathbf{A}[j,j]/D$, $s = \mathbf{A}[i,j]/D$, and $D = \sqrt{\mathbf{A}[j,j]^2 + \mathbf{A}[i,j]^2}$. The special construction of $\mathbf{T}_{i,j}$ implies that the matrix product $\mathbf{T}_{i,j}\mathbf{A}$ transforms $\mathbf{A}$ to a matrix whose (i,j)th element is zero.

Once a general rotation matrix $\mathbf{A}$ is given, its Givens rotations can be found by preparing the Givens rotation matrices $\mathbf{T}_{i,j}$ that zero those elements of $\mathbf{A}$ located below the main diagonal, column by column, from the bottom up. Figure 4.30 is a listing of Matlab code that does that for the rotation matrix that rotates point $(1,1,1)$ to the x axis.

```
n=3;
A=[.5774,-.5774,-.5774; .5774,.7886,-.2115; .5774,-.2115,.7886]
% Rotation from 1,1,1 to x-axis
Q=eye(n);
for j=1:n-1,
  for i=n:-1:j+1,
    T=eye(n);
    D=sqrt(A(j,j)^2+A(i,j)^2);
    cos=A(j,j)/D; sin=A(i,j)/D;
    T(j,j)=cos; T(j,i)=sin; T(i,j)=-sin; T(i,i)=cos; T
    A=T*A;
Q=Q*T';
  end;
end;
Q
A
```

Figure 4.30: Computing Three Givens Matrices.

The three rotation matrices produced by this computation are listed in Figure 4.31, where they are used to rotate point $(1,1,1)$ to the x axis. Matrix `T1` rotates $(1,1,1)$ 45° about the y axis to $(1.4142,1,0)$, which is rotated by `T2` 35.26° about the z axis to $(1.7321,0,0)$, which is trivially rotated by `T3` 15° about the x axis to itself.

```
T1=[0.7071,0,0.7071; 0,1,0; -0.7071,0,0.7071];
T2=[0.8165,0.5774,0; -0.5774,0.8165,0; 0,0,1];
T3=[1,0,0; 0,0.9660,0.2587; 0,-0.2587,0.9660];
p=[1;1;1];
a=T1*p
b=T2*a
c=T3*b
```

Figure 4.31: Rotating Point $(1,1,1)$ to the x Axis.

J. Wallace Givens, Jr. (1910–1993) pioneered the use of plane rotations in the early days of automatic matrix computations. Givens graduated from Lynchburg College in 1928, and he completed his Ph.D. at Princeton University in 1936. After spending three years at the Institute for Advanced Study in Princeton as an assistant of Oswald Veblen, Givens accepted an appointment at Cornell University, but later moved to Northwestern University. In addition to his academic career, Givens was the director of the Applied Mathematics Division at Argonne National Lab and, like his counterpart Alston Householder at Oak Ridge National Laboratory, Givens served as an early president of SIAM. He published his work on the rotations in 1958.

—Carl D. Meyer.

4.4.5 Quaternions

Appendix B is a general introduction to quaternions and should be reviewed before reading ahead. Quaternions can elegantly express arbitrary rotations in three dimensions. Those familiar with complex numbers may have noticed that a rotation in two dimensions is similar to multiplying two complex numbers because the product

$$(a,b)\begin{pmatrix} c & d \\ -d & c \end{pmatrix} = (ac-bd, ad+bc)$$

is identical to the product $(a+ib)(c+id)$. Quaternions extend this similarity to three dimensions as follows. To rotate a point $\mathbf{P}$ by an angle θ about a direction $\mathbf{v}$, we first prepare the quaternion $\mathbf{q} = [\cos(\theta/2), \sin(\theta/2)\mathbf{u}]$, where $\mathbf{u} = \mathbf{v}/|\mathbf{v}|$ is a unit vector in the direction of $\mathbf{v}$. The rotation can then be expressed as the triple product $\mathbf{q}\cdot[0,\mathbf{P}]\cdot\mathbf{q}^{-1}$. Note that our $\mathbf{q}$ is a unit quaternion since $\sin^2(\theta/2)+\cos^2(\theta/2)=1$. This interesting connection between quaternions and rotations is developed in detail in [Hanson 06] (see especially page 50 of this reference).

◇ **Exercise 4.46:** Prove that the triple product $\mathbf{q}\cdot[0,\mathbf{P}]\cdot\mathbf{q}^{-1}$ really performs a rotation of $\mathbf{P}$ about $\mathbf{v}$ (or $\mathbf{u}$). (Hint: Perform the multiplications and show that they produce Equation (4.31).)

As an example of quaternion rotation, consider a 90° rotation of point $\mathbf{P} = (0,1,1)$ about the y axis. The quaternion required is $\mathbf{q} = [\cos 45°, \sin 45°(0,1,0)]$. It is a unit quaternion, so its inverse is $\mathbf{q}^{-1} = [\cos 45°, -\sin 45°(0,1,0)]$. The rotated point is thus

$$\begin{aligned}&\mathbf{q}[0,\mathbf{P}]\mathbf{q}^{-1}\\ &= [-\sin 45°, (\sin 45°, \cos 45°, \cos 45°)]\,[0,(0,1,1)]\,[\cos 45°, -\sin 45°(0,1,0)]\\ &= [0,(1,1,0)].\end{aligned}$$

The quaternion resulting from the triple product always has a zero scalar. We ignore the scalar and find that the point has been moved, by the rotation, from the $x=0$ plane to the $z=0$ plane.

Figure 4.32 illustrates this particular rotation about the y axis and also makes it easy to understand the rule for the direction of the quaternion rotation $\mathbf{q}[0,\mathbf{P}]\mathbf{q}^{-1}$. The rule is: Let $\mathbf{q} = [s,\mathbf{v}]$ be a rotation quaternion in a right-handed three-dimensional

coordinate system. To an observer looking in the direction of $\mathbf{v}$, the triple product $\mathbf{q}[0,\mathbf{P}]\mathbf{q}^{-1}$ rotates point $\mathbf{P}$ clockwise. For a negative rotation angle, the rotation is counterclockwise. In a left-handed coordinate system (Figure 4.32b), the direction of rotation is the opposite.

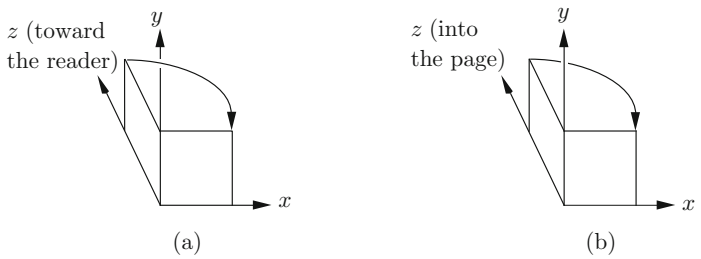


Figure 4.32: Rotation in a Right-Handed (a) and in a Left-Handed (b) Coordinate System.

4.4.6 Concatenating Rotations

Sometimes we have to perform two consecutive rotations on an object. This turns out to be easy and numerically stable with a quaternion representation.

If $\mathbf{q}_1$ and $\mathbf{q}_2$ are unit quaternions representing the two rotations, then associativity of quaternion multiplication implies that the combined rotation of $\mathbf{q}_1$ followed by $\mathbf{q}_2$ is represented by the quaternion $\mathbf{q}_2 \cdot \mathbf{q}_1$. The proof is

$$\mathbf{q}_2 \cdot (\mathbf{q}_1 \cdot \mathbf{P} \cdot \mathbf{q}_1^{-1}) \cdot \mathbf{q}_2^{-1} = (\mathbf{q}_2 \cdot \mathbf{q}_1) \cdot \mathbf{P} \cdot (\mathbf{q}_1^{-1} \cdot \mathbf{q}_2^{-1}) = (\mathbf{q}_2 \cdot \mathbf{q}_1) \cdot \mathbf{P} \cdot (\mathbf{q}_2 \cdot \mathbf{q}_1)^{-1}.$$

Quaternion multiplication involves fewer operations than matrix multiplication, so combining rotations by means of quaternions is faster. Performing fewer multiplications also implies better numerical accuracy.

In general, we use 4×4 transformation matrices to express three-dimensional transformations, so we would like to be able to express the rotation $\mathbf{P}^* = \mathbf{q}[0,\mathbf{P}]\mathbf{q}^{-1}$ as $\mathbf{P}^* = \mathbf{PM}$, where $\mathbf{M}$ is a 4×4 matrix. Given the two quaternions $\mathbf{q}_1 = w_1 + x_1\mathbf{i} + y_1\mathbf{j} + z_1\mathbf{k} = (w_1, x_1, y_1, z_1)$ and $\mathbf{q}_2 = w_2 + x_2\mathbf{i} + y_2\mathbf{j} + z_2\mathbf{k} = (w_2, x_2, y_2, z_2)$, their product is

$$\begin{aligned}\mathbf{q}_1 \cdot \mathbf{q}_2 = {} & (w_1w_2 - x_1x_2 - y_1y_2 - z_1z_2) + (w_1x_2 + x_1w_2 + y_1z_2 - z_1y_2)\mathbf{i} \\ & + (w_1y_2 - x_1z_2 + y_1w_2 + z_1x_2)\mathbf{j} + (w_1z_2 + x_1y_2 - y_1x_2 + z_1w_2)\mathbf{k}.\end{aligned}$$

The first step is to realize that each term in this product depends linearly on the coefficients of $\mathbf{q}_1$. This product can therefore be expressed as

$$\mathbf{q}_1 \cdot \mathbf{q}_2 = \mathbf{q}_2 \cdot \mathbf{L}(\mathbf{q}_1) = (x_2, y_2, z_2, w_2) \begin{pmatrix} w_1 & z_1 & -y_1 & -x_1 \\ -z_1 & w_1 & x_1 & -y_1 \\ y_1 & -x_1 & w_1 & -z_1 \\ x_1 & y_1 & z_1 & w_1 \end{pmatrix}.$$

When $\mathbf{L}(\mathbf{q}_1)$ multiplies the row vector $\mathbf{q}_2$, the result is a row vector representation for $\mathbf{q}_1 \cdot \mathbf{q}_2$. Each term also depends linearly on the coefficients of $\mathbf{q}_2$, so the same product can also be expressed as

$$\mathbf{q}_1 \cdot \mathbf{q}_2 = \mathbf{q}_1 \cdot \mathbf{R}(\mathbf{q}_2) = (x_1, y_1, z_1, w_1) \begin{pmatrix} w_2 & -z_2 & y_2 & -x_2 \\ z_2 & w_2 & -x_2 & -y_2 \\ -y_2 & x_2 & w_2 & -z_2 \\ x_2 & y_2 & z_2 & w_2 \end{pmatrix}.$$

When $\mathbf{R}(\mathbf{q}_2)$ multiplies the row vector $\mathbf{q}_1$, the result is also a row vector representation for $\mathbf{q}_1 \cdot \mathbf{q}_2$.

We can now write the triple product $\mathbf{q} \cdot [0, \mathbf{P}] \cdot \mathbf{q}^{-1}$ in terms of the matrices $\mathbf{L}(\mathbf{q})$ and $\mathbf{R}(\mathbf{q})$:

$$\begin{aligned} \mathbf{q}[0, \mathbf{P}]\mathbf{q}^{-1} &= \mathbf{q}([0, \mathbf{P}] \cdot \mathbf{q}^{-1}) = \mathbf{q}([0, \mathbf{P}]\mathbf{R}(\mathbf{q}^{-1})) \\ &= ([0, \mathbf{P}]\mathbf{R}(\mathbf{q}^{-1}))\mathbf{L}(\mathbf{q}) = [0, \mathbf{P}](\mathbf{R}(\mathbf{q}^{-1})\mathbf{L}(\mathbf{q})) \\ &= [0, \mathbf{P}]\mathbf{M}, \end{aligned}$$

where matrix $\mathbf{M}$ is

$$\begin{aligned} \mathbf{M} &= \mathbf{R}(\mathbf{q}^{-1}) \cdot \mathbf{L}(\mathbf{q}) \\ &= \begin{pmatrix} w & z & -y & x \\ -z & w & x & y \\ y & -x & w & z \\ -x & -y & -z & w \end{pmatrix} \begin{pmatrix} w & z & -y & -x \\ -z & w & x & -y \\ y & -x & w & -z \\ x & y & z & w \end{pmatrix} \\ &= \begin{pmatrix} w^2+x^2-y^2-z^2 & 2xy+2wz & 2xz-2wy & 0 \\ 2xy-2wz & w^2-x^2+y^2-z^2 & 2yz+2wx & 0 \\ 2xz+2wy & 2yz-2wx & w^2-x^2-y^2+z^2 & 0 \\ 0 & 0 & 0 & w^2+x^2+y^2+z^2 \end{pmatrix}. \end{aligned}$$

Since we have unit quaternions, they satisfy $w^2 + x^2 + y^2 + z^2 = 1$, so we can write the final result

$$\mathbf{M} = \begin{pmatrix} 1-2y^2-2z^2 & 2xy+2wz & 2xz-2wy & 0 \\ 2xy-2wz & 1-2x^2-2z^2 & 2yz-2wx & 0 \\ 2xz+2wy & 2yz-2wx & 1-2x^2-2y^2 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}. \tag{4.33}$$

In a left-handed coordinate system, the same rotation is expressed by the triple product $\mathbf{q}^{-1}[0, \mathbf{P}]\mathbf{q}$ or, equivalently, by $\mathbf{P}^* = \mathbf{P} \cdot \mathbf{M}^T$, where $\mathbf{M}^T$ is the transpose of $\mathbf{M}$.

4.5 Transforming the Coordinate System

Our discussion so far has assumed that points are transformed in a static coordinate system. It is also possible (and sometimes useful) to transform the coordinate system instead of the points. To understand the main idea, let's consider the simple example of translation. Suppose that a two-dimensional point $\mathbf{P}$ is transformed to a point $\mathbf{P}^*$ by translating it m and n units in the x and y directions, respectively. How can the transformation be reversed? We consider two ways.

1. Suppose that the original transformation was $\mathbf{P}^* = \mathbf{PT}$, where

$$\mathbf{T} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ m & n & 1 \end{pmatrix}.$$

It is clear that the transformation matrix

$$\mathbf{S} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -m & -n & 1 \end{pmatrix}$$

will transform $\mathbf{P}^*$ back to $\mathbf{P}$. However, it is trivial to show, by using Equation (4.22), that $\mathbf{S}$ is the inverse of $\mathbf{T}$.

2. The transformation can be reversed by translating the coordinate system in the reverse directions (i.e., by $-m$ and $-n$ units) by using an (unknown) transformation matrix $\mathbf{M}$.

Since the two methods produce the same result, we conclude that $\mathbf{M} = \mathbf{S} = \mathbf{T}^{-1}$. Transforming the coordinate axes is therefore done by a matrix that's the inverse of transforming a point. This is true for any affine transformations, not just translation.

Simple kindness to one's self and all that lives is
the most powerful transformational force of all.
—David R. Hawkins.