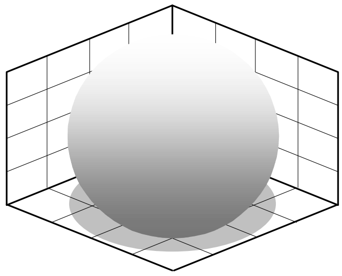
Maths for Computer Graphics



2D transformations

Translation

Cartesian coordinates provide a one-to-one relationship between number and shape.

If $P(x, y)$ is a vertex on a shape

A new point $P'(x', y')$ can be defined using

$$x' = x + 3$$

$$y' = y + 1$$

Where $P'(x', y')$ is three units to the right and one unit above P.

Adding or subtracting a value to or from a coordinate translates it in space.

Maths for Computer Graphics

2D transformations

Translation

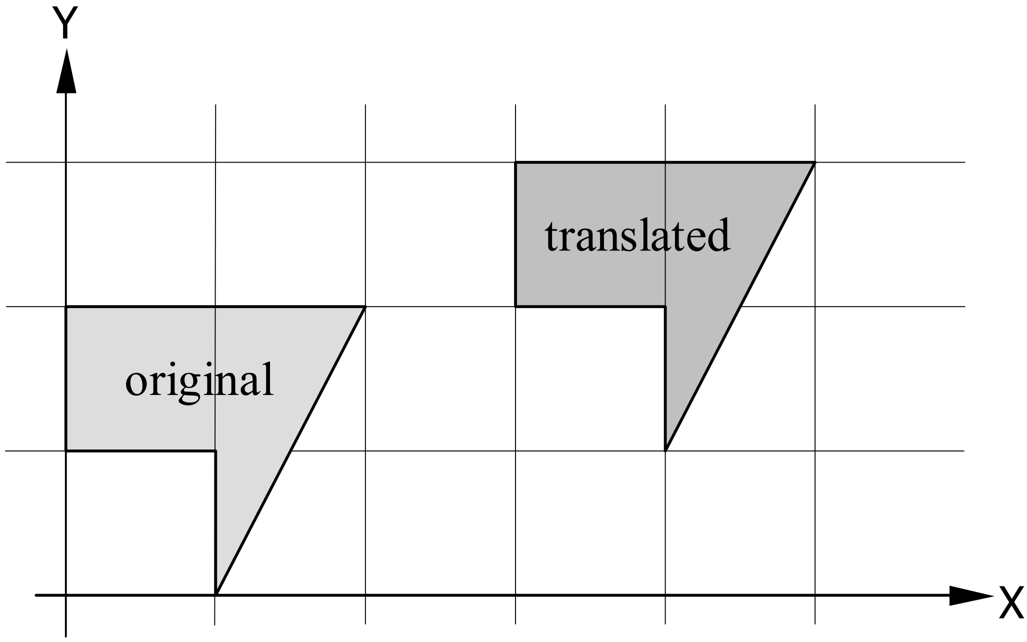


Fig. 7.1 *The translated shape results by adding 3 to every x-coordinate, and 1 to every y-coordinate of the*

The transform is

$$x' = x + 3$$
$$y' = y + 1$$

2D transformations

Scaling

Shape scaling is achieved by multiplying coordinates

$$x' = 2x$$
$$y' = 1.5y$$

This transform results in a horizontal scaling of 2 and a vertical scaling of 1.5.

Note that a point located at the origin does not change its place, therefore, scaling is relative to the origin.

Maths for Computer Graphics

2D transformations

Scaling

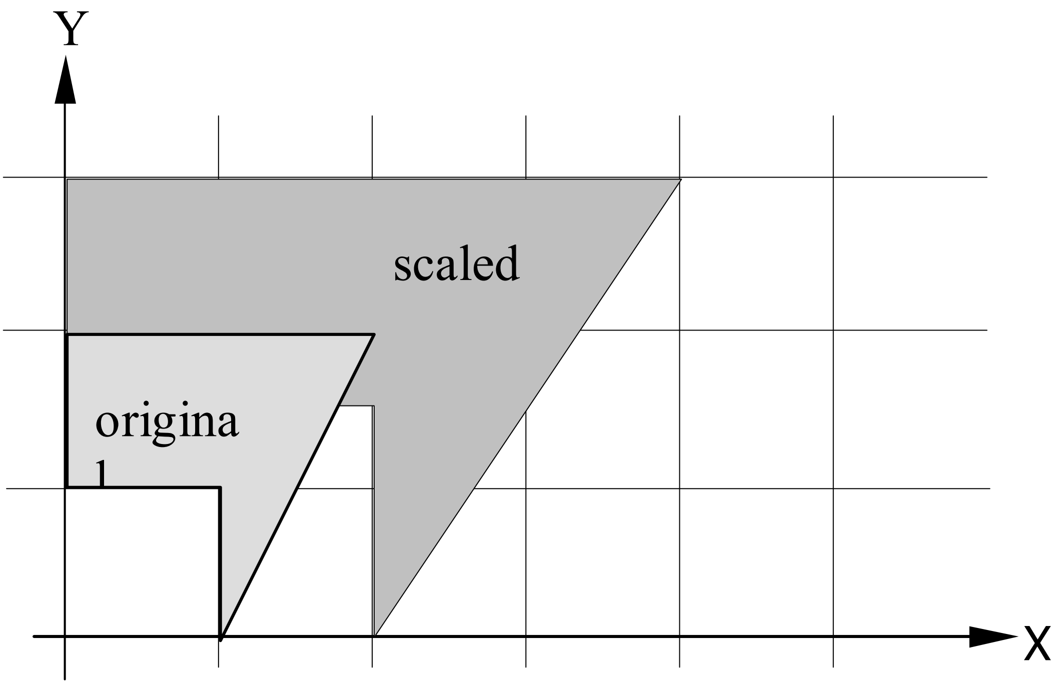


Fig. 7.2 *The scaled shape results by multiplying every x-coordinate by 2 and every y-coordinate by 1.5.*

The transform is

$$x' = 2x$$

$$y' = 1.5y$$

2D transformations

Reflection

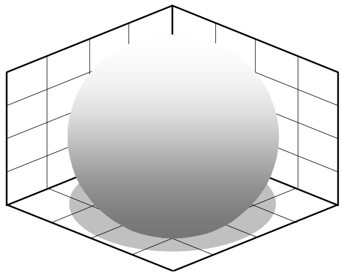
To reflect a shape relative to the *y*-axis

$$x' = -x$$
$$y' = y$$

To reflect a shape relative to the *x*-axis

$$x' = x$$
$$y' = -y$$

Maths for Computer Graphics



2D transformations

Reflection

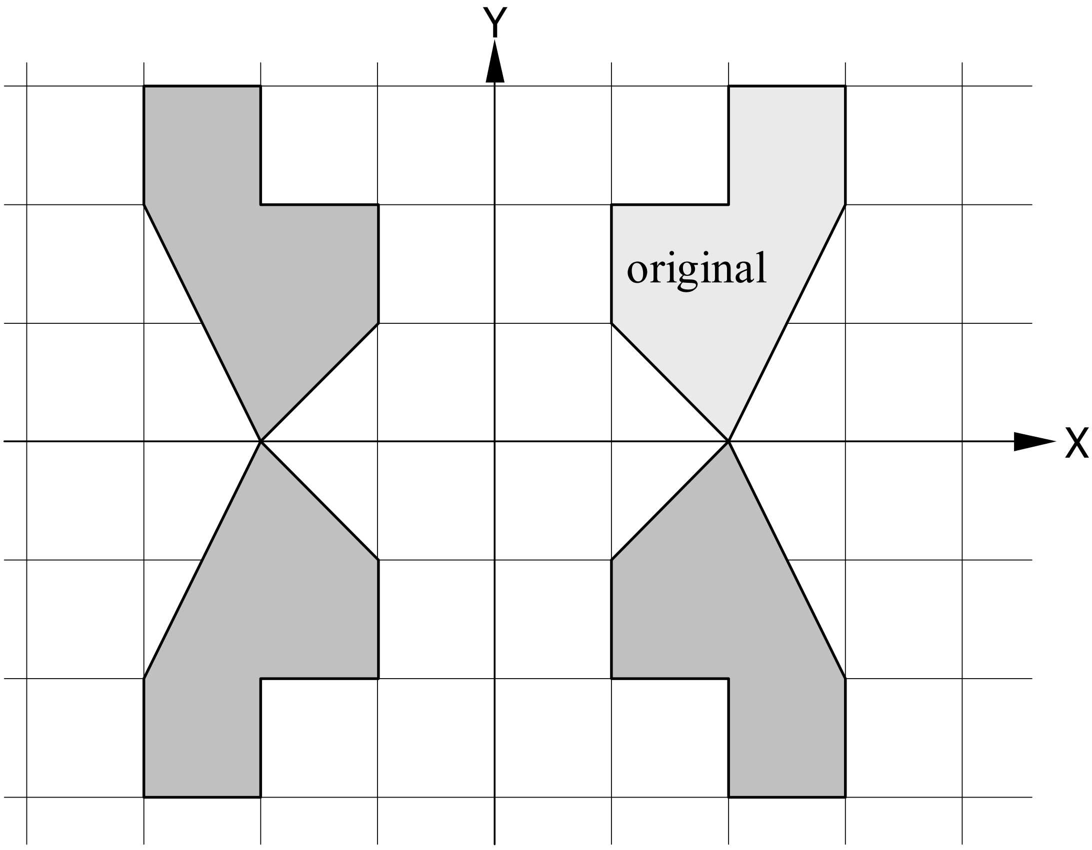


Fig. 7.3 *The top right-hand shape can give rise to the three reflections simply by reversing the signs of* coordinates.

The transform is

$$x' = \pm x$$

$$y' = \pm y$$

Matrices

Matrix notation was investigated by the British mathematician Cayley around 1858.

Caley formalized matrix algebra along with the American mathematicians Benjamin and Charles Pierce.

Carl Gauss (1777-1855) had proved that transformations were not commutative, i.e. $T_1 \times T_2 \neq T_2 \times T_1$, and Caley's matrix notation would clarify such observations.

Consider the transformation T_1

$$T_1 \quad \begin{aligned} x' &= ax + by \\ y' &= cx + dy \end{aligned}$$

Caley proposed separating the constants from the variables as follows

$$T_1 \quad \begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \cdot \begin{bmatrix} x \\ y \end{bmatrix}$$

Maths for Computer Graphics

Matrices

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \cdot \begin{bmatrix} x \\ y \end{bmatrix}$$

Scaling

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} 2 & 0 \\ 0 & 1.5 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \end{bmatrix}$$

Reflection (about the y axis)

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \end{bmatrix}$$

Reflection (about the x axis)

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \end{bmatrix}$$

Translation

?

Homogeneous coordinates

Homogeneous coordinates surfaced in the early 1800s where they were independently proposed by A. F. Möbius (who invented a one-sided curled band), Feuerbach, Étienne Bobillier, and Plücker.

Möbius named them *barycentric coordinates*.

They have also been called *areal coordinates* because of their area calculating properties.

Homogeneous coordinates

Homogeneous coordinates define a point in a plane using three coordinates instead of two.

Initially, Plücker located a homogeneous point relative to the sides of a triangle, but later revised his notation to the one employed in contemporary mathematics and computer graphics.

This states that for a point P with coordinates (x, y) there exists a homogeneous point (xt, yt, t) such that $X = x/t$ and $Y = y/t$.

For example, the point (3, 4) has homogeneous coordinates (6, 8, 2), because 3 = 6/2 and 4 =8/2.

But the homogeneous point (6, 8, 2) is not unique to (3, 4); (12, 16, 4), (15, 20, 5) and (300, 400, 100) are all possible homogeneous coordinates for (3, 4).

Maths for Computer Graphics

Homogeneous coordinates

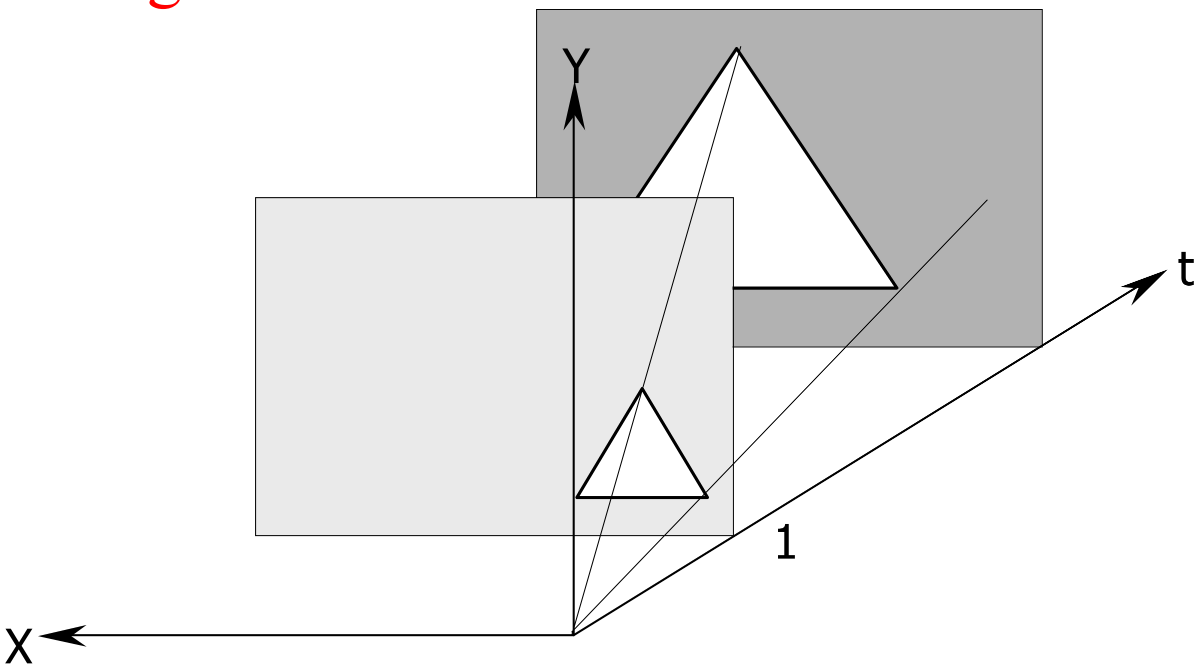
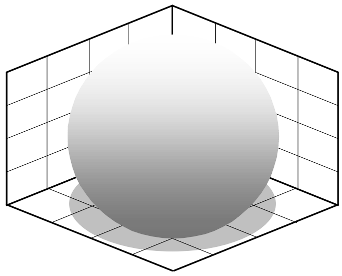


Fig. 7.4 *2D homogeneous coordinates can be visualized as a plane in 3D space, generally where t = 1, for*

Maths for Computer Graphics



Homogeneous coordinates

Consider the following transformation on the homogeneous point $(x, y, 1)$

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} a & b & c \\ d & e & f \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

This expands to

$$x' = ax + by + c$$
$$y' = dx + ey + f$$
$$1 = 1$$

which solves the above problem of translating.

Basically, we ignore the third coordinate 1.

Homogeneous coordinates

2D translation

The algebraic and matrix notation for 2D translation is

$$x' = x + t_x$$

$$y' = y + t_y$$

or using matrices

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix} . \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

e.g.

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 2 \\ 0 & 1 & 3 \\ 0 & 0 & 1 \end{bmatrix} . \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Translates a shape 2 in the x directions and 3 in the y direction.

Homogeneous coordinates

2D scaling

The algebraic and matrix notation for 2D scaling is

$$x' = s_x x$$

$$y' = s_y y$$

or using matrices

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

e.g.

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 1.5 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Scales by a factor of 2 in the x direction and 1.5 in the y direction.

Homogeneous coordinates

2D scaling relative to a point

To scale relative to another point (p_x, p_y):

1: Subtract (p_x, p_y) from (x, y) respectively.
This effectively translates the reference point (p_x, p_y) back to the origin.

2: Perform the scaling operation.

3: Add (p_x, p_y) back to (x, y) respectively, to compensate for the original subtraction.

$$x' = s_x(x - p_x) + p_x$$

$$y' = s_y(y - p_y) + p_y$$

which simplifies to

$$x' = s_x x + p_x(1 - s_x)$$

$$y' = s_y y + p_y(1 - s_y)$$

or in a homogeneous matrix form

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} s_x & 0 & p_x(1-s_x) \\ 0 & s_y & p_y(1-s_y) \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Homogeneous coordinates
2D scaling relative to a point

Example

To scale a shape by 2 relative to the point $(1, 1)$

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 2 & 0 & -1 \\ 0 & 2 & -1 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

$$x' = 2x - 1$$
$$y' = 2y - 1$$

Homogeneous coordinates

2D reflections

The matrix notation for reflecting about the y axis is

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} . \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

or about the x axis

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix} . \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Homogeneous coordinates

2D reflections relative to a line

To make a reflection about an arbitrary vertical or horizontal axis.

e.g. To make a reflection about the vertical axis $x = 1$.

1: Subtract 1 from the x-coordinate.
This effectively makes the $x = 1$ axis coincident with the major y axis.

2: Perform the reflection by reversing the sign of the modified x coordinate.

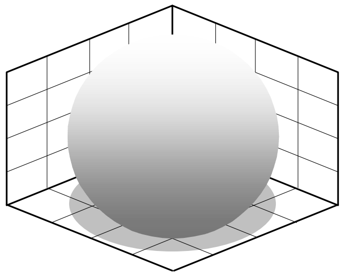
3: Add 1 to the reflected coordinate to compensate for the original subtraction.

$$
\begin{aligned}
x_1 &= x - 1 \\
x_2 &= -(x - 1) \\
x' &= -(x - 1) + 1
\end{aligned}
$$

which simplifies to

$$
\begin{aligned}
x' &= -x + 2 \\
y' &= y
\end{aligned}
$$

Maths for Computer Graphics



Homogeneous coordinates

2D reflections relative to a line

$$x' = -x + 2$$
$$y' = y$$

or in matrix form

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} -1 & 0 & 2 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} . \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

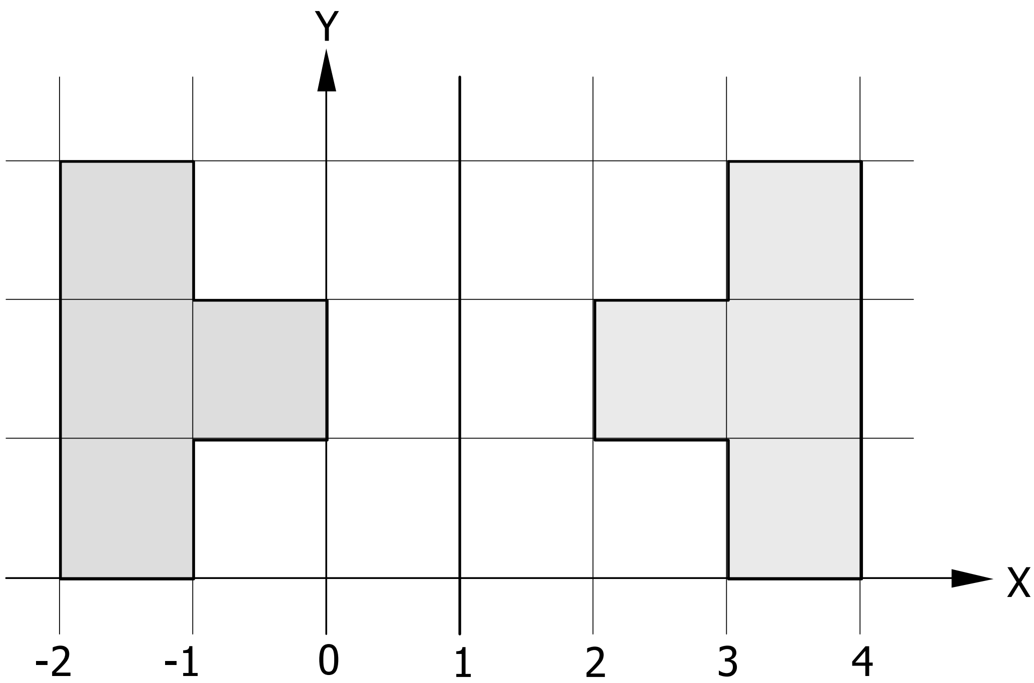


Fig. 7.5 *The shape on the right is reflected*

Homogeneous coordinates

2D reflection relative to a line

A similar transform is used for reflections about an arbitrary x-axis, $y = a_y$

$$x' = x$$
$$y' = -(y - a_y) + a_y \qquad = -y + 2a_y$$

In matrix form

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 2a_y \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Maths for Computer Graphics

2D Shearing

A shape is sheared by leaning it over at an angle β.

The y-coordinate remains unchanged but the x-coordinate is a function of y and tan(β)

$$x' = x + y\tan(\beta)$$
$$y' = y$$

Maths for Computer Graphics

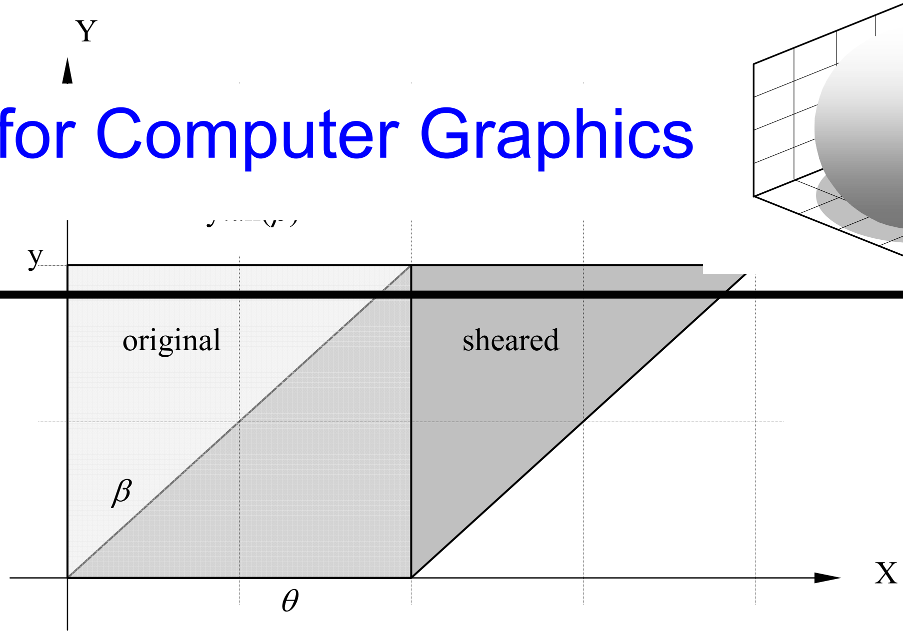


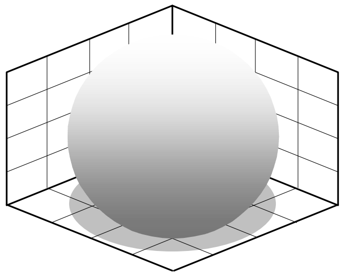
Fig. 7.6 *The original square shape is sheared to the right by and angle β, and the horizontal shift is proportional to ytan(β).*

2D Shearing

In matrix form

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & \tan(\beta) & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} . \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Maths for Computer Graphics



2D Rotation

A point $P(x, y)$ is to be rotated by an angle β about the origin to $P'(x', y')$.

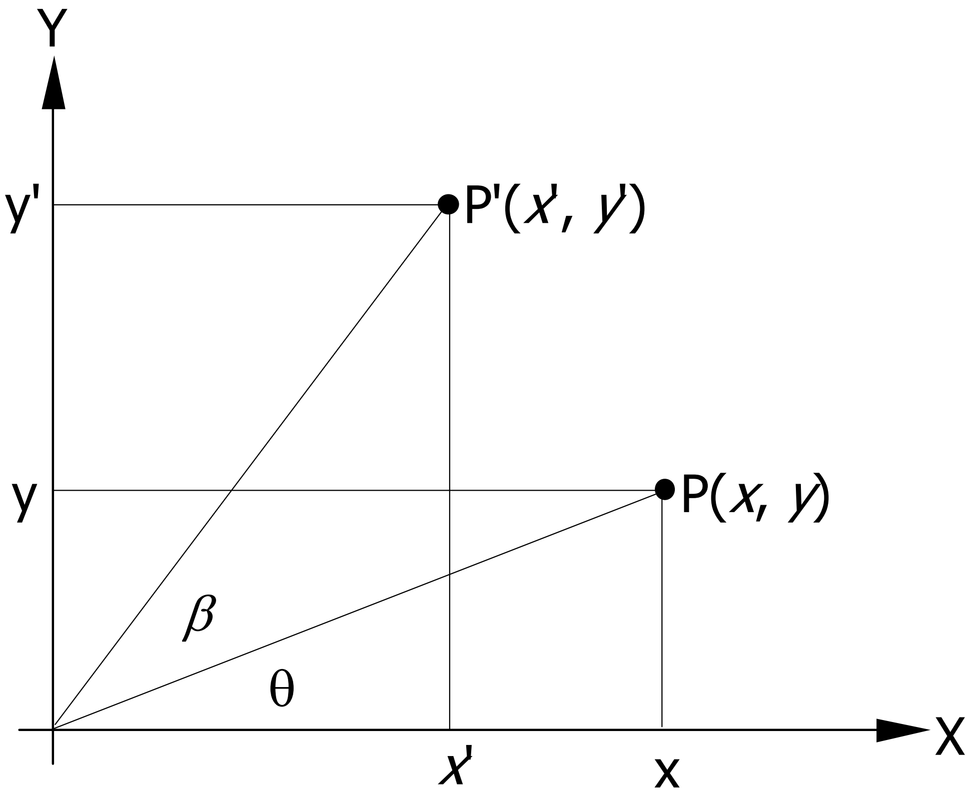
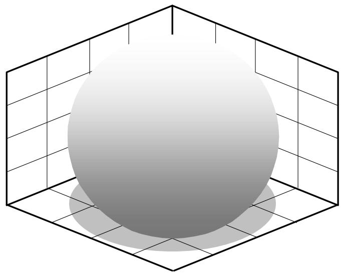


Fig. 7.7 *The point P(x, y) is rotated through an angle β to P′(x′, y′).*

$$x' = R\cos(\theta + \beta)$$

$$y' = R\sin(\theta + \beta)$$

Maths for Computer Graphics



2D Rotation

$$x' = R\cos(\theta+\beta)$$
$$y' = R\sin(\theta+\beta)$$

therefore

$$x' = R\left(\cos(\theta)\cos(\beta) - \sin(\theta)\sin(\beta)\right)$$
$$y' = R\left(\sin(\theta)\cos(\beta) + \cos(\theta)\sin(\beta)\right)$$
$$x' = R\left(\frac{x}{R}\cos(\beta) - \frac{y}{R}\sin(\beta)\right)$$
$$y' = R\left(\frac{y}{R}\cos(\beta) + \frac{x}{R}\sin(\beta)\right)$$
$$x' = x\cos(\beta) - y\sin(\beta)$$
$$y' = x\sin(\beta) + y\cos(\beta)$$

In matrix form

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} \cos(\beta) & -\sin(\beta) & 0 \\ \sin(\beta) & \cos(\beta) & 0 \\ 0 & 0 & 1 \end{bmatrix} . \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

2D Rotation

Example

To rotate a point by 90° the matrix becomes

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} . \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

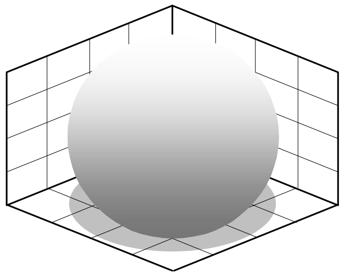
Thus the point (1, 0) becomes (0, 1).

If we rotate by 360° the matrix becomes

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} . \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Such a matrix has a null effect and is called an *identity matrix*.

Maths for Computer Graphics



2D Rotation about an arbitrary point (p_x, p_y)

1: Subtract (p_x, p_y) from the coordinates (x, y).

2: Perform the rotation.

3: Add (p_x, p_y) to the rotated coordinates.

1: Subtract (p_x, p_y)

$$x_1 = (x - p_x)$$

$$y_1 = (y - p_y)$$

2: Rotate β about the origin

$$x_2 = (x - p_x)\cos(\beta) - (y - p_y)\sin(\beta)$$

$$y_2 = (x - p_x)\sin(\beta) + (y - p_y)\cos(\beta)$$

3: Add (p_x, p_y)

$$x' = (x - p_x)\cos(\beta) - (y - p_y)\sin(\beta) + p_x$$

$$y' = (x - p_x)\sin(\beta) + (y - p_y)\cos(\beta) + p_y$$

simplifying

$$x' = x\cos(\beta) - y\sin(\beta) + p_x(1 - \cos(\beta)) + p_y\sin(\beta)$$

$$y' = x\sin(\beta) + y\cos(\beta) + p_y(1 - \cos(\beta)) - p_x\sin(\beta)$$

2D Rotation about an arbitrary point

$$x' = x\cos(\beta) - y\sin(\beta) + p_x(1-\cos(\beta)) + p_y\sin(\beta)$$

$$y' = x\sin(\beta) + y\cos(\beta) + p_y(1-\cos(\beta)) - p_x\sin(\beta)$$

In matrix form

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} \cos(\beta) & -\sin(\beta) & p_x(1-\cos(\beta)) + p_y\sin(\beta) \\ \sin(\beta) & \cos(\beta) & p_y(1-\cos(\beta)) - p_x\sin(\beta) \\ 0 & 0 & 1 \end{bmatrix} . \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

WHICH CAN NOT BE MEMORISED!!

2D Scaling about a point

To scale a point (x, y) relative to some point (p_x, p_y) we:

1: translate $(-p_x, -p_y)$

2: perform the scaling (s_x, s_y)

3: translate (p_x, p_y)

In matrix form

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = [\text{translate}(p_x, p_y)] \cdot [scale(s_x, s_y)] \cdot [\text{translate}(-p_x, -p_y)] \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

which becomes

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & p_x \\ 0 & 1 & p_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & -p_x \\ 0 & 1 & -p_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

2D Scaling about a point

Concatenating

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & p_x \\ 0 & 1 & p_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & -p_x \\ 0 & 1 & -p_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & p_x \\ 0 & 1 & p_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} s_x & 0 & -s_x p_x \\ 0 & s_y & -s_y p_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

and finally

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} s_x & 0 & -s_x p_x \\ 0 & s_y & -s_y p_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

2D reflections about an arbitrary axis

A reflection about an arbitrary axis $x = a_x$, parallel with the y-axis, is given by

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = [translate\ (a_x, 0)] \cdot [reflection] \cdot [translate\ (-a_x, 0)] \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

which expands to

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & a_x \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} . \begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} . \begin{bmatrix} 1 & 0 & -a_x \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} . \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & a_x \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} . \begin{bmatrix} -1 & 0 & a_x \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} . \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} -1 & 0 & 2a_x \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} . \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

2D rotation about an arbitrary point

A rotation about the origin is given by

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} \cos(\beta) & -\sin(\beta) & 0 \\ \sin(\beta) & \cos(\beta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

A rotation about an arbitrary point (p_x, p_y)

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \left[\text{translate}(p_x, p_y)\right] \cdot \left[\text{rotate}\beta\right] \cdot \left[\text{translate}(-p_x, -p_y)\right] \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

which expands to

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & p_x \\ 0 & 1 & p_y \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \cos(\beta) & -\sin(\beta) & 0 \\ \sin(\beta) & \cos(\beta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & -p_x \\ 0 & 1 & -p_y \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} \cos(\beta) & -\sin(\beta) & p_x(1-\cos(\beta)) + p_y \sin(\beta) \\ \sin(\beta) & \cos(\beta) & p_y(1-\cos(\beta)) - p_x \sin(\beta) \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Maths for Computer Graphics

3D transformations

3D translation

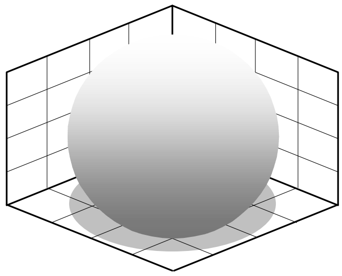
$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & t_x \\ 0 & 1 & 0 & t_y \\ 0 & 0 & 1 & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

3D scaling

$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} s_x & 0 & 0 & 0 \\ 0 & s_y & 0 & 0 \\ 0 & 0 & s_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

3D scaling relative to a point

$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} s_x & 0 & 0 & p_x(1-s_x) \\ 0 & s_y & 0 & p_y(1-s_y) \\ 0 & 0 & s_z & p_z(1-s_z) \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$



3D Rotations

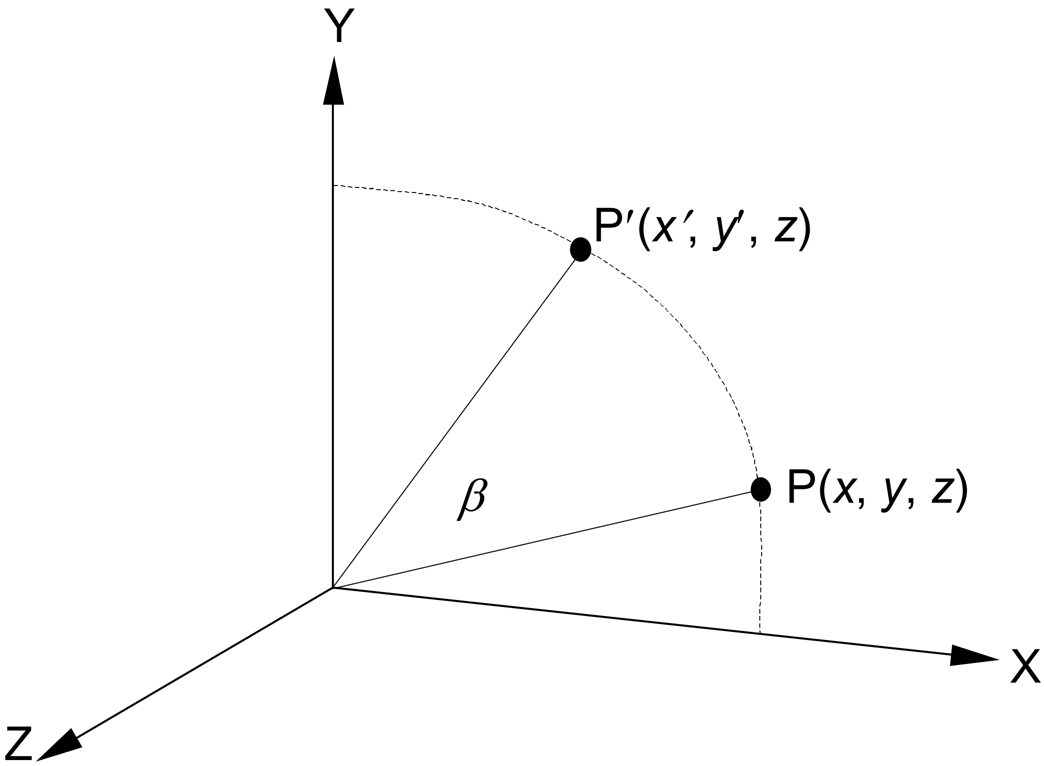


Fig. 7.8 *Rotating P about the z-axis.*

$$x' = x\cos(\beta) - y\sin(\beta)$$
$$y' = x\sin(\beta) + y\cos(\beta)$$
$$z' = z$$

$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} \cos(\beta) & -\sin(\beta) & 0 & 0 \\ \sin(\beta) & \cos(\beta) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

3D Rotations

When rotating about the x-axis, the x-coordinate remains constant whilst the y- and z-coordinates are changed. Algebraically, this is

$$
\begin{aligned}
x' &= x \\
y' &= y\cos(\beta) - z\sin(\beta) \\
z' &= y\sin(\beta) + z\cos(\beta)
\end{aligned}
$$

In a matrix transform

$$
\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\beta) & -\sin(\beta) & 0 \\ 0 & \sin(\beta) & \cos(\beta) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}
$$

3D Rotations

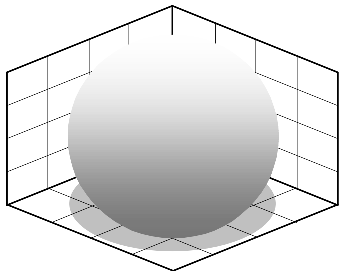
When rotating about the y-axis, the y-coordinate remains constant whilst the x- and z-coordinates are changed. Algebraically, this is

$$x' = z\sin(\beta) + x\cos(\beta)$$
$$y' = y$$
$$z' = z\cos(\beta) - x\sin(\beta)$$

In matrix form

$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} \cos(\beta) & 0 & \sin(\beta) & 0 \\ 0 & 1 & 0 & 0 \\ -\sin(\beta) & 0 & \cos(\beta) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

Maths for Computer Graphics



Yaw, Pitch and Roll rotations

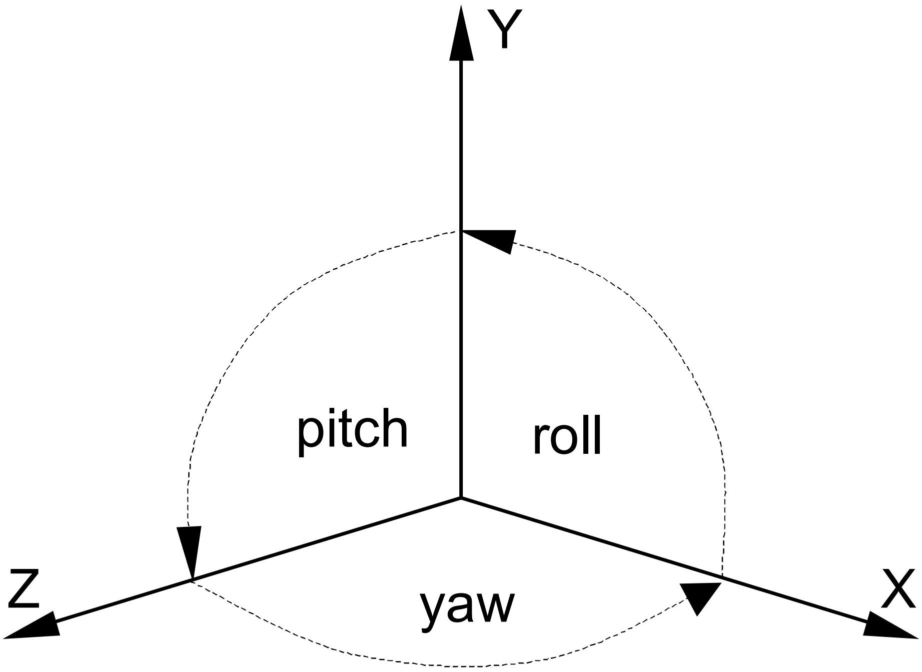


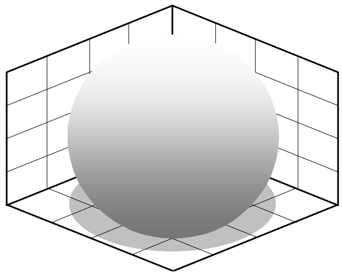
***Fig. 7.9** The convention for roll, pitch and yaw*

Roll about the z axis

Pitch about the x axis

Yaw about the y axis

Maths for Computer Graphics



Roll, Pitch and Yaw Euler rotations

rotate *roll* about the *z*-axis

$$\begin{bmatrix} \cos(roll) & -\sin(roll) & 0 & 0 \\ \sin(roll) & \cos(roll) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

rotate *pitch* about the *x*-axis

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(pitch) & -\sin(pitch) & 0 \\ 0 & \sin(pitch) & coa(pitch) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

rotate *yaw* about the *y*-axis

$$\begin{bmatrix} \cos(yaw) & 0 & \sin(yaw) & 0 \\ 0 & 1 & 0 & 0 \\ -\sin(yaw) & 0 & \cos(yaw) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Combined rotations

A common way of combining yaw, pitch and roll is

$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = [yaw] \cdot [pitch] \cdot [roll] \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

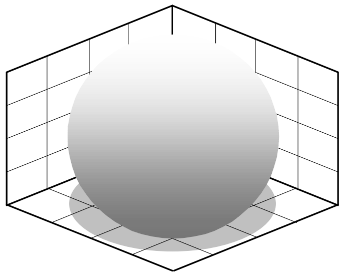
If translation is involved

$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = [translate] \cdot [yaw] \cdot [pitch] \cdot [roll] \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

The vertex is first rotated about the z-axis (roll), followed by a rotation about the x-axis (pitch), followed by a rotation about the y-axis (yaw), then translated.

Euler rotations are relative to the fixed frame of reference.

Maths for Computer Graphics



Gimbal Lock

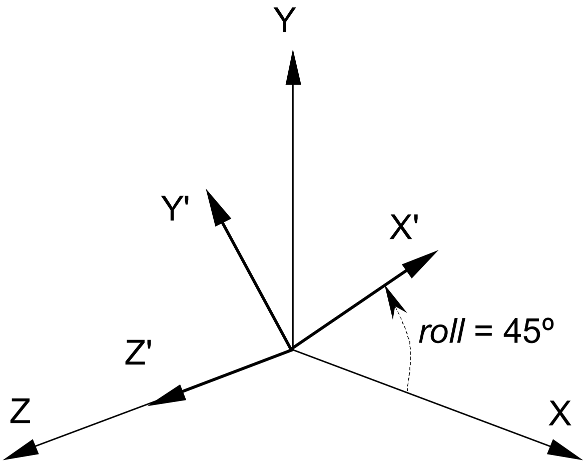


Fig. 7.12 *The X'Y'Z' axial system after a roll of 45°.*

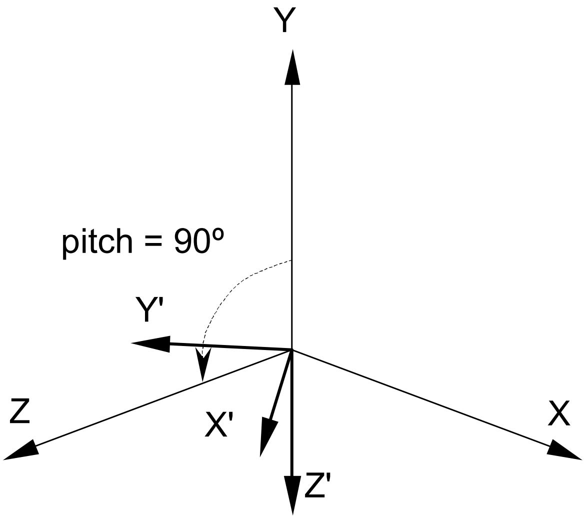


Fig. 7.13 *The X'Y'Z' axial system after a pitch of 90°.*

Figure 7.12 shows the orientation of $X'Y'Z'$ after it is subjected to a roll of 45° about the z-axis.

Figure 7.13 shows the orientation of $X'Y'Z'$ after it is subjected to a pitch of 90° about the x-axis.

If we now performed a yaw of 45° about the z-axis, it would rotate the x'-axis towards the x-axis, counteracting the effect of the original roll.

Effectively, a yaw has become a negative roll rotation, caused by the 90° pitch. This situation is known as *gimbal lock*, because one degree of rotation has been lost.

Maths for Computer Graphics

Rotating about an axis

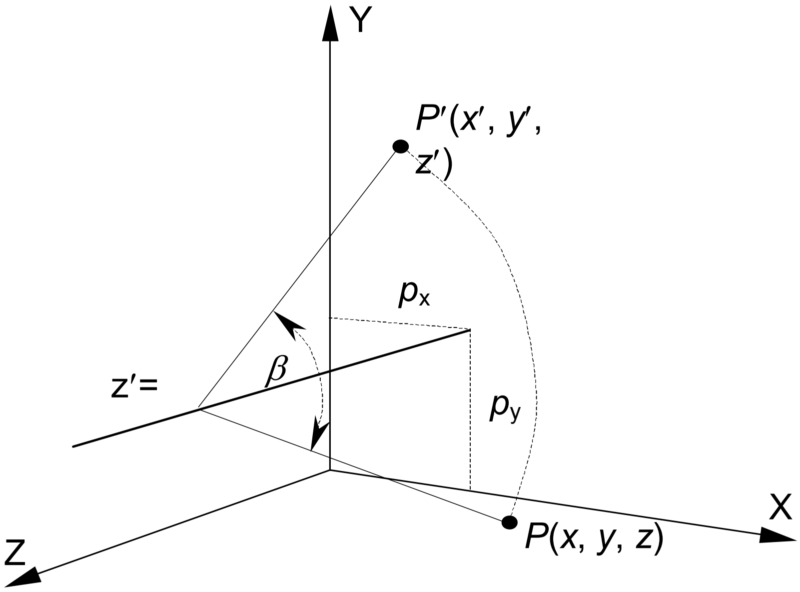


Fig. 7.14 *Rotating a point about an axis parallel with the z-axis.*

$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = [translate(p_x, p_y, 0)] \cdot [rotate\beta] \cdot [translate(-p_x, -p_y, 0)] \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

Or in matrix form

$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} \cos(\beta) & -\sin(\beta) & 0 & p_x(1-\cos(\beta)) + p_y \sin(\beta) \\ \sin(\beta) & \cos(\beta) & 0 & p_y(1-\cos(\beta)) - p_x \sin(\beta) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

Determinants

Determinants arise in the solution of linear equations

$$c_1 = a_1 x + b_1 y \qquad (1)$$

$$c_2 = a_2 x + b_2 y \qquad (2)$$

Solving for x

Multiply (1) by b_2 and (2) by b_1

$$c_1 b_2 = a_1 b_2 x + b_1 b_2 y$$

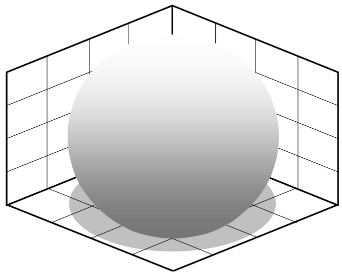
$$c_2 b_1 = a_2 b_1 x + b_1 b_2 y$$

Then subtract

$$c_1 b_2 - c_2 b_1 = a_1 b_2 x - a_2 b_1 x$$

$$x = \frac{c_1 b_2 - c_2 b_1}{a_1 b_2 - a_2 b_1}$$

Provided that the denominator $a_1 b_2 - a_2 b_1 \neq 0$.



Determinants

$$c_1 = a_1 x + b_1 y \qquad (1)$$

$$c_2 = a_2 x + b_2 y \qquad (2)$$

Solving for y

Multiply (1) by a_2 and (2) by a_1

$$a_2 c_1 = a_1 a_2 x + a_2 b_1 y$$

$$a_1 c_2 = a_1 a_2 x + a_1 b_2 y$$

Subtracting

$$a_1 c_2 - a_2 c_1 = a_1 b_2 y - a_2 b_1 y$$

$$y = \frac{a_1 c_2 - a_2 c_1}{a_1 b_2 - a_2 b_1}$$

Provided that the denominator $a_1 b_2 - a_2 b_1 \neq 0$.

Determinants

$$x = \frac{c_1 b_2 - c_2 b_1}{a_1 b_2 - a_2 b_1}$$

$$y = \frac{a_1 c_2 - a_2 c_1}{a_1 b_2 - a_2 b_1}$$

Let

$$\begin{vmatrix} a_1 & b_1 \\ a_2 & b_2 \end{vmatrix} = a_1 b_2 - a_2 b_1 \text{ = the determinat}$$

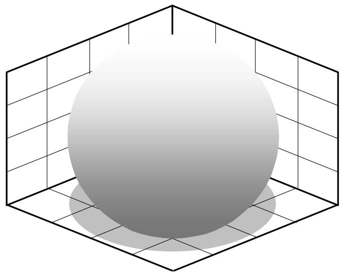
$$\frac{x}{c_1 b_2 - c_2 b_1} = \frac{1}{a_1 b_2 - a_2 b_1}$$

$$\frac{y}{a_1 c_2 - a_2 c_1} = \frac{1}{a_1 b_2 - a_2 b_1}$$

$$\frac{x}{c_1 b_2 - c_2 b_1} = \frac{y}{a_1 c_2 - a_2 c_1} = \frac{1}{a_1 b_2 - a_2 b_1}$$

Et voila!

$$\frac{x}{\begin{vmatrix} c_1 & b_1 \\ c_2 & b_2 \end{vmatrix}} = \frac{y}{\begin{vmatrix} a_1 & c_1 \\ a_2 & c_2 \end{vmatrix}} = \frac{1}{\begin{vmatrix} a_1 & b_1 \\ a_2 & b_2 \end{vmatrix}}$$



Determinants summary

$$c_1 = a_1 x + b_1 y$$

$$c_2 = a_2 x + b_2 y$$

$$\begin{bmatrix} c_1 \\ c_2 \end{bmatrix} = \begin{bmatrix} a_1 & b_1 \\ a_2 & b_2 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

$$\frac{x}{\begin{vmatrix} c_1 & b_1 \\ c_2 & b_2 \end{vmatrix}} = \frac{y}{\begin{vmatrix} a_1 & c_1 \\ a_2 & c_2 \end{vmatrix}} = \frac{1}{\begin{vmatrix} a_1 & b_1 \\ a_2 & b_2 \end{vmatrix}}$$

Example

$$10 = 2x + y$$

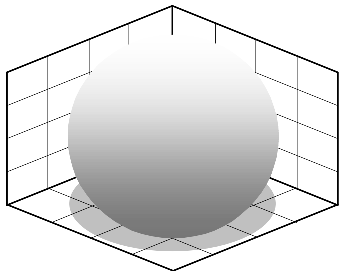
$$4 = 5x - y$$

$$\frac{x}{\begin{vmatrix} 10 & 1 \\ 4 & -1 \end{vmatrix}} = \frac{y}{\begin{vmatrix} 2 & 10 \\ 5 & 4 \end{vmatrix}} = \frac{1}{\begin{vmatrix} 2 & 1 \\ 5 & -1 \end{vmatrix}}$$

$$\frac{x}{-14} = \frac{y}{-42} = \frac{1}{-7}$$

Therefore $x = 2$ and $y = 6$

Maths for Computer Graphics



Determinants

With a set of three linear equations:

$$d_1 = a_1x + b_1y + c_1z$$
$$d_2 = a_2x + b_2y + c_2z$$
$$d_3 = a_3x + b_3y + c_3z$$

the value of x is defined as

$$x = \frac{d_1b_2c_3 - d_1b_3c_2 + d_2b_3c_1 - d_2b_1c_3 + d_3b_1c_2 - d_3b_2c_1}{a_1b_2c_3 - a_1b_3c_2 + a_2b_3c_1 - a_2b_1c_3 + a_3b_1c_2 - a_3b_2c_1}$$

with similar expressions for y and z.

Once more, the denominator comes from the determinant of the matrix associated with the matrix formulation of the linear equations:

Determinants

$$d_1 = a_1x + b_1y + c_1z$$
$$d_2 = a_2x + b_2y + c_2z$$
$$d_3 = a_3x + b_3y + c_3z$$

$$\begin{bmatrix} d_1 \\ d_2 \\ d_3 \end{bmatrix} = \begin{bmatrix} a_1 & b_1 & c_1 \\ a_2 & b_2 & c_2 \\ a_3 & b_3 & c_3 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \end{bmatrix}$$

therefore

$$\frac{x}{\begin{vmatrix} d_1 & b_1 & c_1 \\ d_2 & b_2 & c_2 \\ d_3 & b_3 & c_3 \end{vmatrix}} = \frac{y}{\begin{vmatrix} a_1 & c_1 & d_1 \\ a_2 & c_2 & d_2 \\ a_3 & c_3 & d_3 \end{vmatrix}} = \frac{z}{\begin{vmatrix} a_1 & b_1 & d_1 \\ a_2 & b_2 & d_2 \\ a_3 & b_3 & d_3 \end{vmatrix}} = \frac{1}{\begin{vmatrix} a_1 & b_1 & c_1 \\ a_2 & b_2 & c_2 \\ a_3 & b_3 & c_3 \end{vmatrix}}$$

$$\begin{vmatrix} a_1 & b_1 & c_1 \\ a_2 & b_2 & c_2 \\ a_3 & b_3 & c_3 \end{vmatrix} = a_1b_2c_3 - a_1b_3c_2 + a_2b_3c_1 - a_2b_1c_3 + a_3b_1c_2 - a_3b_2$$

which can be written

$$a_1\begin{vmatrix} b_2 & c_2 \\ b_3 & c_3 \end{vmatrix} - a_2\begin{vmatrix} b_1 & c_1 \\ b_3 & c_3 \end{vmatrix} + a_3\begin{vmatrix} b_1 & c_1 \\ b_2 & c_2 \end{vmatrix}$$

Graphical interpretation of the determinant

Consider the transform

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \cdot \begin{bmatrix} x \\ y \end{bmatrix}$$

The determinant of the transform is $ad - cb$.

The vertices of the unit-square are moved as follows

$$(0, 0) \Rightarrow (0, 0)$$
$$(1, 0) \Rightarrow (a, c)$$
$$(1, 1) \Rightarrow (a + b, c + d)$$
$$(0, 1) \Rightarrow (b, d)$$

Determinant and the unit square

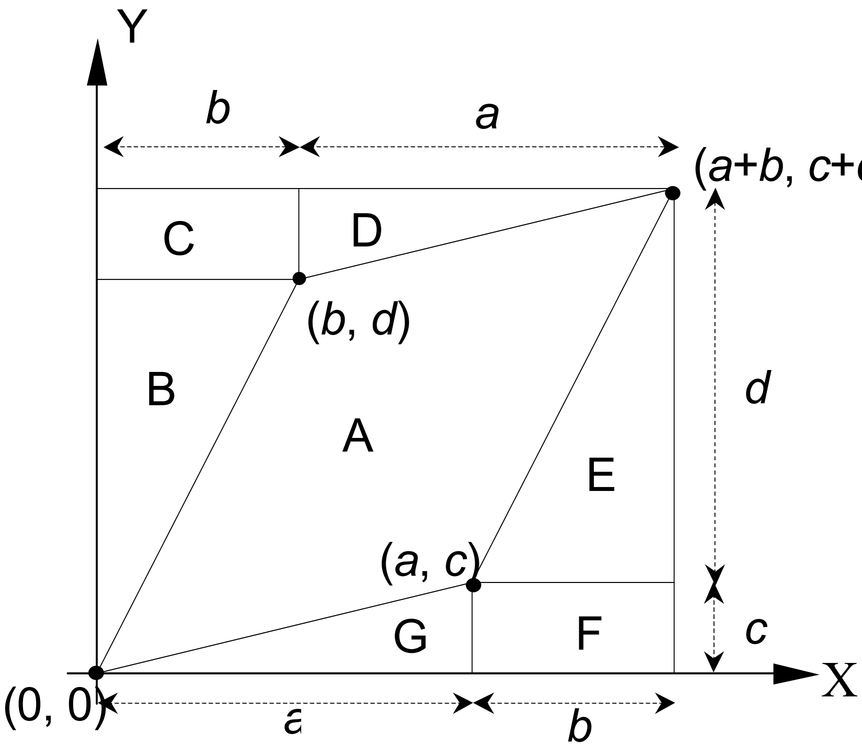


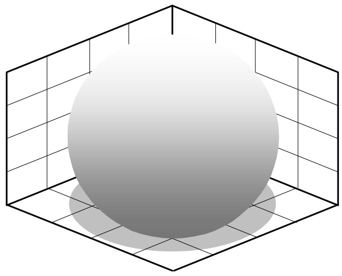
***Fig. 7.28** The inner parallelogram is the transformed unit square*

$$\begin{aligned} area &= (a+b)(c+d) - \mathrm{B} - \mathrm{C} - \mathrm{D} - \mathrm{E} - \mathrm{F} - \mathrm{G} \\ &= ac + ad + cb + bd - \tfrac{1}{2}bd - cb - \tfrac{1}{2}ac - \tfrac{1}{2}bd - cb - \tfrac{1}{2}ac \\ &= ad - cb \end{aligned}$$

which is the determinant of the transform.

But as the area of the original unit-square was 1, the determinant of the transform controls the scaling factor applied to the transformed shape.

Maths for Computer Graphics



Determinant of a transform

This transform encodes a scaling of 2, and results in an overall area scaling of 4

$$\begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} \text{and the determinant is} \begin{vmatrix} 2 & 0 \\ 0 & 2 \end{vmatrix} = 4$$

This transform encodes a scaling of 3 and a translation of (3, 3), and results in an overall area scaling of 9:

$$\begin{bmatrix} 3 & 0 & 3 \\ 0 & 3 & 3 \\ 0 & 0 & 1 \end{bmatrix} \text{and the determinant is}$$

$$3\begin{vmatrix} 3 & 3 \\ 0 & 1 \end{vmatrix} - 0\begin{vmatrix} 0 & 3 \\ 0 & 1 \end{vmatrix} + 0\begin{vmatrix} 0 & 3 \\ 3 & 3 \end{vmatrix} = 9$$

Consider the rotate transform

$$\begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} \text{its determinant is } \cos^2(\theta) + \sin^2(\theta) = 1$$