

Homogeneous Coordinates

Interestingly we can use the extra dimension in homogeneous coordinates to distinguish a point from a vector. A point will have a 1 in the last component, and a vector will have a 0. The difference between a point and a vector is a bit wish washy in my mind so I'm not sure why this distinction helps.

Transforming a Point

Say we have the 2D point (x, y) . This point as a column vector in homogeneous coordinates is $\begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$. For a multiplication between this vector and a transformation matrix (3 by 3) to work we need to do the matrix times the vector (in that order) to give the translated vector, $Av = v'$.

Combining Transformations

Say we want to do a translation then a rotation (A then B) on the point x. First we must do $Ax = x'$, then $B(x')$. That is BAx . The order is important as matrix multiplication is not commutative, i.e. $AB \neq BA$ (just think a translation then a rotation is not necessarily the same as a rotate then a move (by the same amounts)). If we do lots of transformations we may get something like $DCBAx$, this is in effect doing transformation A then B then C then D. (Remember matrix multiplication is associative, i.e. $(AB)C = A(BC)$).

As a side note, if you express your point as a row vector (eg. $(x \ y \ z \ 1)$), then to do a transformation you must do xA (where x is the point/row vector). In this case $xABC$ is equivalent to doing transformation A on point x, then transformation B then C (apparently this is how DirectX works).

Affine Transformations

Affine transformations are a special kind of transformation. They have a matrix form where the last row is $[0 \dots 0 \ 1]$. An affine transformation is equivalent to a linear transformation followed by a translation. That is, is the same as $\vec{y} = A\vec{x} + \vec{b}$.

Something interesting to note is, the inverse transformation of an affine transformation is another affine transformation, whose matrix is the inverse matrix of the original. Also an affine transformation in 2D is uniquely defined by its action on three points.

From page 209 of the text (Hill, 2006), affine transformations have some very useful properties.

1. Affine Transformations Preserve Affine Combinations of Points

For some affine transformation T, points P_1 and P_2 , and real's a_1 and b_1 where $a_1 + b_1 = 1$,
 $T(a_1P_1 + a_2P_2) = a_1T(P_1) + a_2T(P_2)$

2. Affine Transformations Preserve Lines and Planes

That is under any affine transformation lines transformed are still lines (they don't suddenly become curved), similarly planes that are transformed are still planes.

3. Parallelism of Lines and Planes is Preserved

"If two lines or planes are parallel, their images under an affine transformation are also parallel." The explanation that Hill uses is rather good,

Take an arbitrary line $A + \mathbf{b}t$ having direction \mathbf{b} . It transforms to the line given in homogeneous coordinates by $M(A + \mathbf{b}t) = MA + (M\mathbf{b})t$, this transformed line has direction vector $M\mathbf{b}$. This new direction does not depend on point A. Thus two different lines $A_1 + \mathbf{b}t$ and $A_2 + \mathbf{b}t$ that have the same direction will transform into two lines both having the direction $M\mathbf{b}$, so they are parallel. The same argument can be applied to planes and beyond.

4. The Columns of the Matrix Reveal the Transformed Coordinate Frame

Take a generic affine transformation matrix for 2D,

$$M = \begin{pmatrix} a & b & c \\ d & e & f \\ 0 & 0 & 1 \end{pmatrix}$$

$$\mathbf{m}_1 = \begin{pmatrix} a \\ d \\ 0 \end{pmatrix} \quad \mathbf{m}_2 = \begin{pmatrix} b \\ e \\ 0 \end{pmatrix}$$

The first two columns, $\begin{pmatrix} a \\ d \\ 0 \end{pmatrix}$ and $\begin{pmatrix} b \\ e \\ 0 \end{pmatrix}$, are vectors (last component is 0). The last

column $\begin{pmatrix} c \\ f \\ 1 \end{pmatrix}$ is a point (last component is a 1).

$$\mathbf{i} = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}, \quad \mathbf{j} = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}, \quad \phi = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$$

Using the standard basis vectors to the vector \mathbf{m}_1 . $\mathbf{m}_1 = M\mathbf{i}$. Similarly for \mathbf{j} and ϕ .

5. Relative Ratios are Preserved

6. Area's Under an Affine Transformation

Given an affine transformation as a matrix M,

$$\frac{\text{area after transformation}}{\text{area before transformation}} = |\det M|$$

7. Every Affine Transformation is Composed of Elementary Operations

Every affine transformation can be constructed by a composition of elementary operations (see below). That is,

$$M = (\text{shear})(\text{scale})(\text{rotation})(\text{translation})$$

For a 2D affine transformation M. In 3D,

$$M = (\text{scale})(\text{rotation})(\text{shear}_1)(\text{shear}_2)(\text{translation})$$

Rotations

Euler's theorem: Any rotation (or sequence of rotations) about a point is equivalent to a single rotation about some coordinate axis through that point. Pages 221-223 of Hill give a detailed explanation of this, as well as the equations to go from one form to the other.

W2V (Window to Viewport Mapping)

A simplified OpenGL pipeline applies the modelview matrix, projection matrix, clipping, then the viewport matrix. The viewport matrix is the window to viewport map.

The window coordinate system is somewhere on the projection plane. These coordinates need to be mapped to the viewport (the area on the screen)

References

F.S. Hill, et al. (2006). Computer Graphics using OpenGL. Second Ed.

Source: <http://andrewharvey4.wordpress.com/2009/08/23/comp3421-lec-2-transformations/>