MATH 111: Application of Matrix Algebra to Computer Graphics

*Yu Yue: 07635165*

*XIAO Sa: 07592272*

Computer graphics are graphics represented on a screen by manipulating some data stored in computer. It’s not difficult to find that the mathematics used in computer graphics is intimately connected with matrix algebra. Some important applications of matrix algebra to computer graphics will be discussed in the rest of the article.

1. *Homogeneous Coordinates*

The homogeneous coordinates of a point (1, 2, 3, … , n) in n  are usually written as the point (1, 2, 3, … , n，) in n + 1 with ≠ 0. We say that (1, 2, 3, … , n) has homogeneous coordinates (1, 2, 3, … , n，) with ≠ 0. For example, the 3D point (x, y, z) can be represented as (x, y, z, 1) in homogeneous coordinates, while (x, y, z, w) with w ≠ 0 can be transferred to be 3D point (x/w, y/w, z/w) again.

Homogeneous coordinates is commonly used in computer graphics because it makes the complex transformations of points solved by combining individual simple ones of matrix multiplication. For instance, any linear transformation on 2 is represented with respect to homogeneous coordinates by a partitioned matrix of the form, where A is a 2 × 2 matrix:

1. A translation of the form (x, y) (x + h, y +k) is written in homogeneous coordinates as (x, y, 1) (x + h, y +k, 1), which can be computed by matrix multiplication:

=

1. A scaling of (x, y, 1) in homogeneous coordinates to be (sx, ty, 1):

=

1. A reflection of (x, y, 1) in homogeneous coordinates to be (y, x, 1):

=

1. A counterclockwise rotation of (x, y, 1) about the origin, angle :

=

The composition of above matrix multiplication, which corresponds to different kinds of transformation, makes the movement of a figure much simpler.

1. *3D Homogeneous Coordinates*

By the definition of homogeneous coordinates, (X, Y, Z, H) are homogeneous coordinates for (x, y, z) with H 0 and x = X / H, y = Y / H, and z = Z / H. In particular, (X, Y, Z, 1) are normalized homogeneous coordinates for the point (x, y, z) in iff H = 1, x = X / H, y = Y / H, and z = Z / H. Hence, each nonzero scalar multiple of (x, y, z, 1) gives a set of homogeneous coordinates for (x, y, z). For example, both (2, 2, 2, 2) and (1, 1, 1, 1) are homogeneous coordinates for (1, 1, 1, 1).

Similarly, it’s easy to make the transformation of 3D case by analogy with the 2D case mentioned before:

1. A translation of the form (x, y, z) (x + h, y + k, z + m) is written in homogeneous coordinates as (x, y, z, 1) (x + h, y + k, z + m, 1), which can be computed by matrix multiplication:

=

1. Rotation about the x-axis through an angle of α:

=

1. Rotation about the y-axis through an angle of β:

=

1. Rotation about the z-axis through an angle of :

=

1. *Example for 3D homogeneous coordinates:*

Give 4 × 4 matrices for the following transformations:

1. Rotation about the y-axis through an angle of 30°.

Solution:

Since a rotation in 3D is a linear transformation, we can construct the 3×3 matrix for the rotation, that is.

Since the 4th dimension of the homogeneous coordinate neither affects nor gets affected by the rotation, the rotation matrix for homogeneous coordinates is

1. Translation by vector (-6, 4, 5).

Solution:

To translate, we keep the 3×3 matrix as the identity matrix. To get the offset (-6, 4, 5), the translation matrix for homogeneous coordinates is.

1. *Perspective Projections*

A three-dimensional object is represented on the two-dimension computer screen by projection the object onto a viewing plane. A perspective projection maps each point (x, y, z) onto an image point (x\*, y\*, 0). The center of projection which resembles the eye position is located at (0, 0, d). The perspective projection must satisfy the condition that the three points (x, y, z), (x\*, y\*, 0) and (0, 0, d) are collinear.

From the collinear relation among the point on actual object (x, y, z), its projection point (x\*, y\*, 0) and the center of projection (0, 0, d), we can derive x\* = x / (1 - z/d) and y\* = y / (1 - z/d).

Now we are able to represent the perspective projection *P* by a matrix using homogeneous coordinates, i.e. mapping (x, y, z, 1) into (x / (1 - z/d), y / (1 - z/d), 0, 1).

Since (x / (1 - z/d), y / (1 - z/d), 0, 1) is equivalent to (x, y, 0, 1 - z/d) in homogeneous coordinates, we map (x, y, z, 1) into (x, y, 0, 1 - z/d) instead:

*P* = =.

1. *Example for perspective projections:*

Find the image of *S*, the box with vertices (3,1,5), (5,1,5), (5,0,5), (3,0,5), (3,1,4), (5,1,4), (5,0,4), (3,0,4), under the perspective projection with center of projection at (0, 0, 10).

Solution:

Let *P* be the projection matrix and *D* be the data matrix for *S* using homogeneous coordinates. Then the data matrix for the image of *S* is

*PD* = =

Transforming the homogeneous coordinates in the data matrix for the image of *S* into 3 coordinates we get

In summary, a typical computer graphics transformation on 3 is represented with respect to homogeneous coordinates by a 4×4 matrix of the form

Where *A* is a 3×3 matrix that produces a linear transformation (rotation, shearing or scaling), **p** is a 3 vector that translates points and **q** is a 3 vector associated with a perspective transformation, and *r* is a scalar (usually 1).

The reasons we take advantage of 4D homogeneous coordinates and 4×4 matrices for 3D graphics modeling and rendering are:

1. Translation cannot be performed by 3×3 matrix because it is not a linear transformation. This motivation brought 4D homogeneous coordinates and 4×4 matrices into use. Now we can perform any affine transformation  
   .
2. The 4D homogeneous coordinates provides a way to represent points at infinity (by setting the 4th dimension w = 0).
3. The vector **q** provides non-affine transformation for perspective projection.

*Reference:* <http://en.wikipedia.org/wiki/Homogeneous_coordinates#Use_in_computer_graphics>

Linear Algebra and Its Applications, by David C. Lay