Introduction to Graphics

Lecture 4:

More Transforms and Homogeneous Coordinates

Lecture Overview

- Affine Transformations
- Reflections
- Mirrors
- Shear
- Homogeneous Coordinates (as vectors)
- Transformation Matrices
- Change of Axes

Affine Transformations

- Affine transformations preserve parallel lines
- Most of the transformations we require are affine:
  - Scaling (enlarge/shrink)
  - Translating (moving)
  - Rotating (turning)
- More complex transformations can be built from these
- Non-affine transformations:
  - perspective projection

Reflections

- Reflections are a very useful special effect that can be easily programmed
- The reflection transformation is affine, and requires the definition of a reflection plane:

$$\mathbf{n}.\mathbf{p} = k$$

where **n** is the normal vector

Reflection

If the reflection plane lines up with the cartesian axes

x=0 or y=0 or z=0

Then the transformation is trivial:

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

YZ Reflection Plane — XZ Reflection Plane — XY Reflection Plane

Mirrors

- For reflection in an infinite plane, a whole object can be transformed and then added to the data base.
- Reflections in a finite mirror is similar to a window onto a 3D graphics scene.

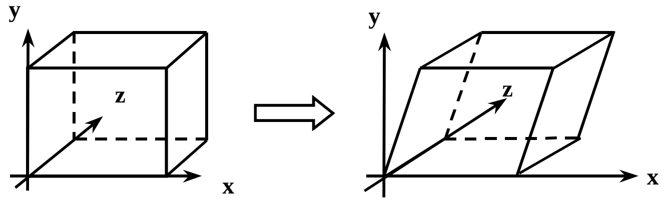
Shear Transformation

- Shear transformations are of use in object deformation and possibly morphing.
- Shears are introduced to an object or a scene by using an asymmetric transformation matrix

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

Simple shears are affine

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



Shearing in the z direction

A similar affine shear is produced when a=0 and b is non zero

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

Non-affine shears

- In general if both a and b are non-zero:

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

the result is a more complex distortion in which the parallel lines are destroyed.

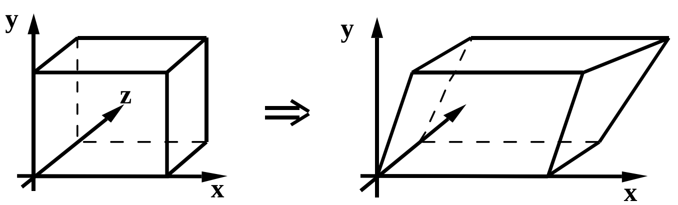
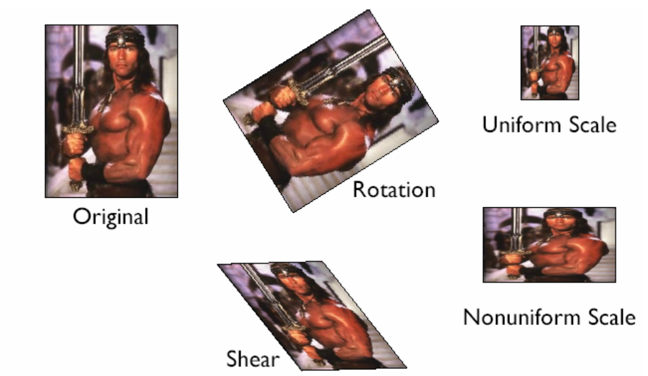


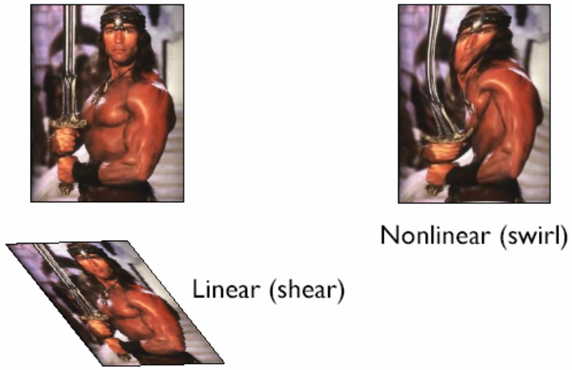
Diagram 4.1 The shear transformation

Some Examples

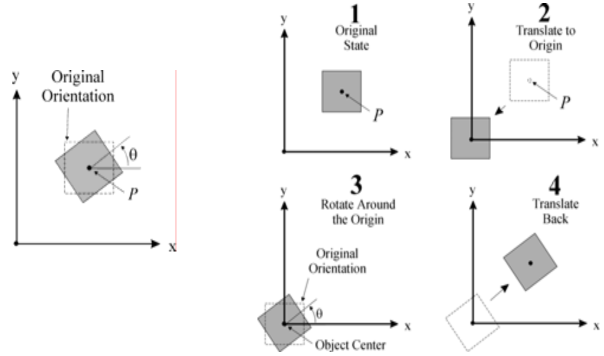


Images from *Conan The Destroyer*, 1984

Linear vs Nonlinear



Standard Transformation Composition



Homogenous coordinates as vectors

- Look again at homogeneous coordinates, and their relation to vectors.
- In the previous lecture we described the fourth ordinate as a scale factor.

[X, Y, Z, h] is equivalent to [X/h, Y/h, Z/h]

Homogeneous Cartesian

Homogenous coordinates and positions

- Consider a normalised homogenous coordinate:

[X,Y,Z,1]

to be equivalent to ***a position vector*** **P**.

Homogenous coordinates and directions

- If the last ordinate is zero:

  [x,y,z,0]

  the coordinate cannot be normalised.

- A homogenous coordinate of this form cannot be associated with a point in Cartesian space.
- However, it still contains information about the relative sizes of x, y and z. Hence we can consider it to be ***a direction vector* d**.

Homogenous coordinates and vectors

In summary, there are two types of homogenous coordinates:

1. Those with the final ordinate non-zero, which can be normalised into *position vectors*.
2. Those with zero in the final ordinate which are *direction vectors* and have direction magnitude.

Vector Addition

If we add two direction vectors, the result is also a direction vector:

[xi,yi,zi,0] + [xj,yj,zj,0] = [xi+xj, yi+yj, zi+zj,0]

This is the normal vector addition rule.

Adding position and direction vectors

If we add a direction vector to a position vector we obtain a position vector:

[Xi,Yi,Zi,1]+[xj,yj,zj,0] = [Xi+xj,Yi+yj,Zi+zj,1]

This result, ties in with our definition of a straight line in cartesian space being defined by a point and a direction:

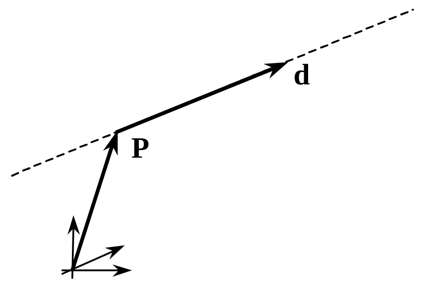


Diagram 4.2 Adding a direction vector to a position vector

Adding two position vectors

If we add two position vectors we obtain their mid-point:

[Xi,Yi,Zi,1] + [Xj,Yj,Zj,1] = [Xi+Xj,Yi+Yj,Zi+Zj,2]
= [(Xi+Xj)/2,(Yi+Yj)/2,(Zi+Zj)/2,1]

This is a reasonable result since adding two position vectors has no meaning in vector algebra!

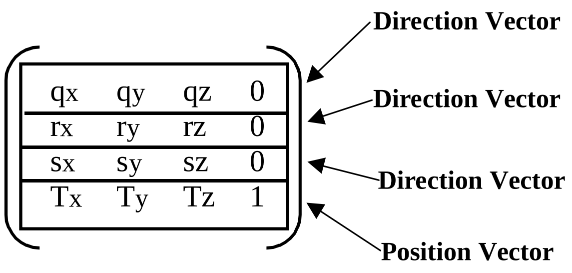


Diagram 4.3 The composition of an affine transformation matrix

Characteristics of Transformation matrices

- In a direction vector the zero in the last ordinate ensures vectors will not be affected by the translation.
- In a position vector the 1 in the last ordinate means all vectors will be moved by the same factor.
- If we do not shear the object the three vectors **q**, **r** and **s** will remain orthogonal, i.e:

q • **r** = **r** • **s** = **q** • **s** = 0.

What do the individual rows mean?

Consider the effect of the transformation in simple cases.

Take the unit vectors along the Cartesian axes, e.g. along the x axis, i = [1,0,0,0]

$$[1,0,0,0] \begin{pmatrix} qx & qy & qz & 0 \\ rx & ry & rz & 0 \\ sx & sy & sz & 0 \\ Tx & Ty & Tz & 1 \end{pmatrix} = [qx, qy, qz, 0]$$

Axis Transformation

Similarly, direction

j = [0,1,0,0]

will be transformed to direction

[rx,ry,rz,0]

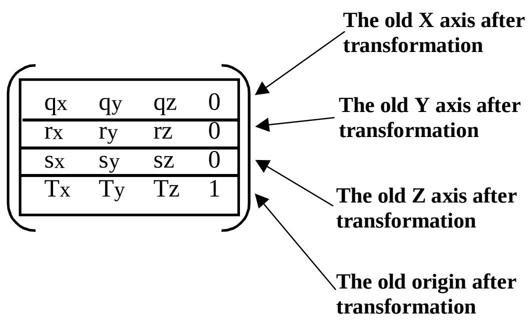
and k = [0,0,1,0]

will be transformed to [sx,sy,sz,0].

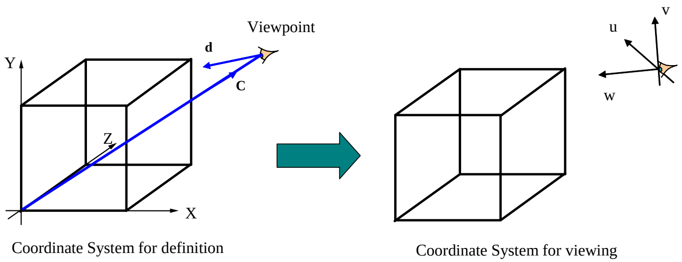
Transforming the Origin

$$[0,0,0,1] \begin{pmatrix} qx & qy & qz & 0 \\ rx & ry & rz & 0 \\ sx & sy & sz & 0 \\ Tx & Ty & Tz & 1 \end{pmatrix} = [Tx, Ty, Tz, 1]$$

Meaning of a transformation matrix



Change of Axes using the Dot Product



Given [u,v,w] and C, find the transformation matrix that moves the scene to that coordinate system.

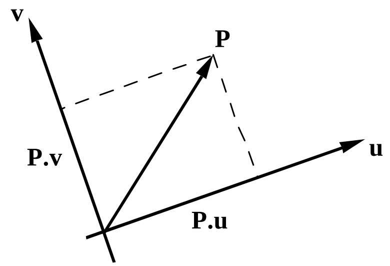


Diagram 4.4
The dot product as a projection

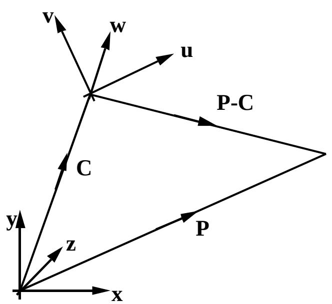


Diagram 4.5
Change of axes using the dot product

Transforming point P

Given point P in the [x,y,z] axis system, we can calculate the corresponding point in the [u,v,w] space as:

P'x = (**P-C**).***u*** **= P.*u* - C.*u***
P'y = (**P-C**).***v*** **= P.*v* - C.*v***
P'z = (**P-C**).***w*** **= P.*w* - C.*w***

Or in Matrix form:

$$[P'x, P'y, P'z, 1] = [Px, Py, Pz, 1] \begin{pmatrix} ux & vx & wx & 0 \\ uy & vy & wy & 0 \\ uz & vz & wz & 0 \\ -C\bullet u & -C\bullet v & -C\bullet w & 1 \end{pmatrix}$$

An Example

The required viewing coordinate system for a graphics scene is defined by:

u = [0,0,-1] ***v*** = [0,1,0] ***w*** = [-1,0,0]

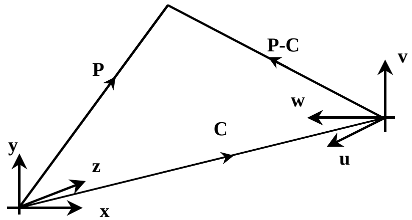
With the viewpoint at:

C = [20, 10, -5]

1. What is the coordinate of the point P = [10,15,5] in the viewing coordinate system?

2. What is the transformation matrix?

Solution



P = [10,15,5] **C** = [20,10,-5] so **P-C**= [-10,5,10]

(**P-C**).***u*** = (**P-C**).[0,0,-1] = -10

(**P-C**).***v*** = (**P-C**).[0,1,0] =5

(**P-C**).***w*** = (**P-C**).[-1,0,0] =10

Hence the coordinate is [10,5,-10]

Solution 2

C.*u* = [20,10,-5] . [0,0,-1] = 5

C.*v* = [20,10,-5] . [0,1,0] = 10

C.*w* = [20,10,-5] . [-1,0,0] = -20

So the transformation matrix is:

$$\begin{pmatrix} 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ -5 & -10 & 20 & 1 \end{pmatrix}$$