Harry Langford
hjel2@cam.ac.uk

Graphics Supervision 2

Model Transform

Changing the object in object coordinates to the object in world coordinates – scaling, rotating and translating etc The time you need to use object coordinates are texture mapping. finding the intersection of a ray with an object. In world coordinates the camera can be anywhere. In world coordinates you are just constructing the scene and converting everything into the same coordinate system so that you can say where everything is. There is one model transform per object.

View Transform

This transforms everything from the world coordinates so that the camera is at the origin in these coordinates and everything moves respective to the camera. This is just so that you make the maths easier. There is one transform per camera.

Projection Transformation

Projecting the 3D coordinates onto a 2D plane. Perspective projection is the most common, however there are other projections such as ones which just go "first in the z-axis" one transform per camera

You can precompute the view and projection transformations for the whole scene – chaining these together means you have to do significantly less work.

Perspective transformations scale down the vector to the point so that they are just all on the same plane but that the unit vector in their direction is unchanged.

With perspective transformations you don't map the z onto the plane – you map it to $\frac{1}{z}$ so that you can know which order things should be in. This is called the "z-buffer".

Matrices

To try and figure out what a complicated matrix does is to throw in some arbitrary points, see what happens. After that's done, throw in the corners of a unit square etc and get an intuition for roughly what the matrix is doing.

There is a very limited set of transformations that a matrix could be doing (rotation, scaling, transformation, projection) and it's usually obvious which it is. For example if there are no $\sqrt{}$'s in the matrix then there is no rotation. If one of the right hand rows is non-zero then there is a transformation, if the bottom row (except w) is nonzero then there is a projection.

In graphics, dividing by a basis vector almost always means projection.

MIPmap's

Bilinear interpolation interpolates in the x-y axis while Trilinear interpolation is bilinear interpolation between pixels in a MIPmap plus interpoltion between two adjacent layers of a MIPmap.

All a MIPMap is is a cache of precomputed downsampled values which can be looked up in constant time rather than computed for each pixel in quadratic time.

If a question asks how do you do a thing – you do need to say manually how to do it.