Computer Graphics: Projection
Z Sweedyk
Lecture 7

Graphics Pipeline 1
1. Build Primitives (Model Coordinates)

Polygon in Model Coordinates
Vertices:
\[ v_1 = (0,0,0), v_2 = (0,1,0), v_3 = (1,1,0), v_4 = (1,0,0) \]

Polygon in Homogenous Model Coordinates
Vertices:
\[ v_1 = (0,0,0,1), v_2 = (0,1,0,1), v_3 = (1,1,0,1), v_4 = (1,0,0,1) \]

Graphics Pipeline 2
1. Build Primitives (Model Coordinates)
2. Assemble Scene (World Coordinates)

Polygon in Homogeneous World Coordinates
For each vertex \( v_i \):
\[ M_1M_2M_3v_i = u_i \]

Scale by \((10,5,5)\), Rotate -45° about y axis, Translate by \((5,2,-10)\)
For the moment we'll ignore clipping.

Project into 2d: What does the viewer see?

Depends on:
- Type of Projection
- Viewpoint
- Viewing Volume

Orthographic Projection

- Viewpoint Default:  
  - Position: (0,0,∞)  
  - Direction: (0,0,1)  
  - Up (0,1,0)
Orthographic Projection (Near Plane at z=0)

Points inside viewing volume project to near plane
\((x,y,z) \rightarrow (x,y,0)\)

Points outside viewing volume are not seen.

Orthographic Projection Matrix

\[
\begin{bmatrix}
1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1
\end{bmatrix}
\]

Projection (near plane at z=0)

- Orthographic
  \((x,y,z,1) \rightarrow (x,y,0,1)\)
- Perspective
  \((x,y,z,1) \rightarrow (?,?,0,?)\)

A second use for that extra dimension

Perspective Projection

- Viewpoint Default:  
  - Position: \((0,0,\text{eye})\)
  - Direction: \((0,0,-1)\)
  - Up \((0,1,0)\)
- Viewing Volume

Perspective Projection: frustum

Projection (near) plane corners: 
(left top, 0) and (right bottom, 0)
Viewpoint \((0,0,\text{eye})\)

Far plane at \(z = -\text{far}\)
Perspective Projection

• Viewpoint Default:
  - Position: (0,0,eye)
  - Direction: (0,0,-1)
  - Up (0,1,0)
• Viewing Volume
  - Projection (near)
    Plane corners:
    (left,top,0),
    (right,bottom,0)
  - Far plane at z=-far

Perspective Projection

\[(x,y,z) \rightarrow (?,?,?)\]

This is just the intersection of a line and plane problem

Perspective Projection

Line: \[P + \alpha (Q-P)\]
Plane: normal \(n\) and point \(S\)
\[R: P + \alpha (Q-P) \text{ for } \alpha = \frac{(P-S)\cdot n}{(P-Q)\cdot n}\]

Corner: \(S\)
Vertex: \(Q\)

Viewpoint: \(P\)

Normal \(n\)

Graphics Pipeline 4

1. Build Primitives (Model Coordinates)
2. Assemble Scene (World Coordinates)
3. Project into 2D (Projected Coordinates)
4. Normalize Normalized Coordinates
Normalize

\[
\begin{bmatrix}
  x \\
  y \\
  z \\
  w
\end{bmatrix}
\rightarrow
\begin{bmatrix}
  x/w \\
  y/w \\
  z/w \\
  1
\end{bmatrix}
\]

Graphics Pipeline 5

1. Build Primitives (Model Coordinates)
2. Assemble Scene (World Coordinates)
3. Project into 2D (Projected Coordinates)
4. Normalize (Normalized Coordinates)
5. Scan Convert with Hidden Surface Removal