Oblique and Perspective Projections

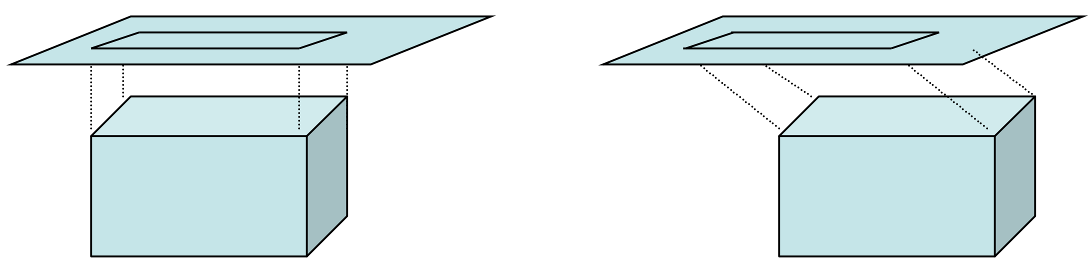
CS116A
Chris Pollett
Nov. 29, 2004.

Outline

- Oblique Parallel Projections
- Perspective projections
- 3D Screen coordinates

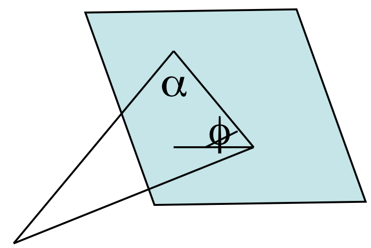
Oblique Projections

- In a parallel projection, if the the lines of projection are not perpendicular to the viewing plane the projection is called oblique.
- For example, the right figure is being projected obliquely.



Drafting and Design

- In engineering and architecture, an oblique projection is often specified by giving two angles: α and ϕ.



- A point A= (x, y, z) maps under an oblique projection to B= (xp, yp, zvp).
- Let C be (x,y, zvp)
- Then α is the angle ABC
- ϕ is the angle between the line L from B to C and the horizontal line of view plane.

Some Equations

- So xp = x + L cos ϕ, yp = x + L sin ϕ
- tan α = (zvp -z)/ L.
- That is, L = (zvp -z)*cot α
- Let L1= cot α. This equals L, where zvp-z =1
- So can write:

  xp = x+L1(zvp-z)cos ϕ, yp = y+L1(zvp-z)sin ϕ

  This is an orthogonal projection when L1=0.
- Notice this is a shearing transformation in the z-axis

Cavalier and Cabinet Parallel Projections

- Typical choices for ϕ are 30 or 45 degrees.
- tan α is usually chosen to be 1 or 2.
- The (45,1) case is called a **cavalier projection** (lines perp to viewing axis retain their length)
- The (30,2) case is called a **cabinet projection** (lines perp to viewing axis half their length)

Oblique Parallel Projection Vector

- In graphics packages that support oblique projections the direction of projection to the view plane is specified with a parallel projection vector Vp from some particular point (x,y,z) to (xp,yp, zvp). So Vpy/Vpx = tan ϕ
- From this get (xp-x)/(zvp -z) = Vpx/Vpy
- Also, get (yp-y)/(zvp -z) = Vpy/Vpz

More Equations

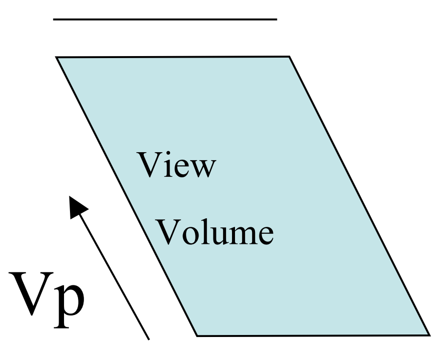
- So x and y transform to:

  $xp = x + (zvp - z)Vpx/Vpz$ and

  $yp = y + (zvp - z)Vpy/Vpz$

Clipping Window and Oblique Parallel-Projection View Volume

- View volume is set up in a similar fashion to orthogonal case.
  - Specify a clipping window
  - Have a near and far plane



Oblique Parallel Projection Matrix

- The matrix look like M_oblique:

$$\begin{bmatrix} 1 & 0 & -Vpx/Vpz & zvpVpx/Vpz \\ 0 & 1 & -Vpy/Vpz & zvpVpy/Vpz \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Normalization Transformation for an Oblique Parallel Projection

- Once we have done our projection, we want to map things into our normalized cube
- To do this we compose M_ortho,norm with M_oblique

Perspective Transformation Coordinates

- To do a perspective transformation need to specify a perspective reference point in (xprp, yprp, zprp).
- Points along line from (x,y,z) to this perspective point given by:
- $x' = x - (x-xprp)u$, $y' = y - (y-yprp)u$ and $z' = z - (z-zprp)u$ where u is between 0 and 1.

Calculating where things go

- So u =(zvp -z)/(zprp - z)
- Substituting this back get
- xp = x*(zprp-zvp)/(zprp-z) + xprp*u and yp= y*(zprp-zvp)/(zprp-z) +yprp*u