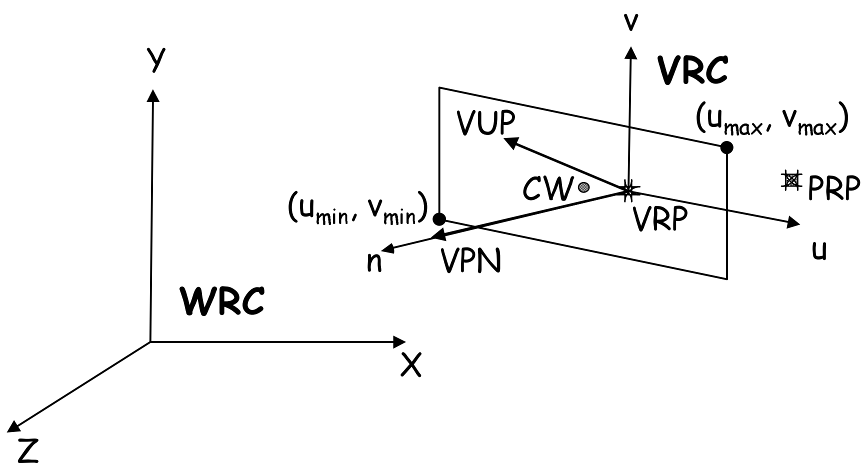
3D Projection and Clipping

But first… Test #1

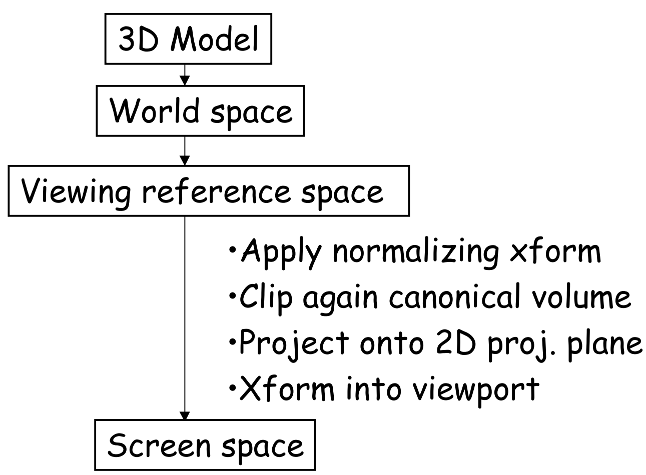
- Average 65.0
- Median 66.0
- Std. Dev 18.2

<40	40-49	50-59	60-69	70-79	80-89	90+
4	5	6	12	11	13	3

Specifying a view



3D viewing process, again



Normalizing Transformation for Perspective Views

1. Translate VRP to origin
2. Rotate the VRC system so that VPN become z-axis, u become x-axis and v become y-axis
3. Translate so that the CoP given by the PRP is at origin
4. Shear such that the center line of the view volume becomes the z-axis
5. Scale so that the view volume becomes the canonical view volume

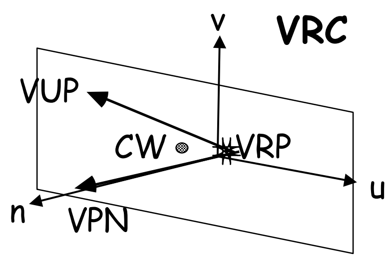
1. Translate VRP to origin

$$\begin{pmatrix} 1 & 0 & 0 & -VRP_X \\ 0 & 1 & 0 & -VRP_y \\ 0 & 0 & 1 & -VRP_Z \\ 0 & 0 & 0 & 1 \end{pmatrix} = \mathbf{T(-VRP)}$$

2. Rotate VRC

We want to take u into (1, 0, 0)
v into (0, 1, 0)
n into (0, 0, 1)

First derive n, u, and v from user input:



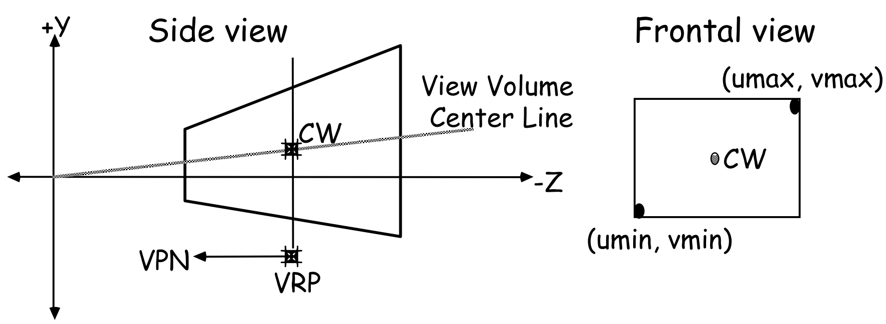
2. Rotate VRC (cont.)

$$\begin{pmatrix} u_x & u_y & u_z & 0 \\ v_x & v_y & v_z & 0 \\ n_x & n_y & n_z & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} = R_{VRC}$$

3. Translate PRP to the origin

$$\begin{pmatrix} 1 & 0 & 0 & -PRP_u \\ 0 & 1 & 0 & -PRP_v \\ 0 & 0 & 1 & -PRP_n \\ 0 & 0 & 0 & 1 \end{pmatrix} = \mathbf{T(-PRP)}$$

4. Shear such that the center line of the view volume becomes the z-axis



Direction of projection (DoP) = CW - PRP

The center line of the view volume is DoP

Shear (cont.)

Multiply DoP with a matrix to get $(0,0,DoP_z)$

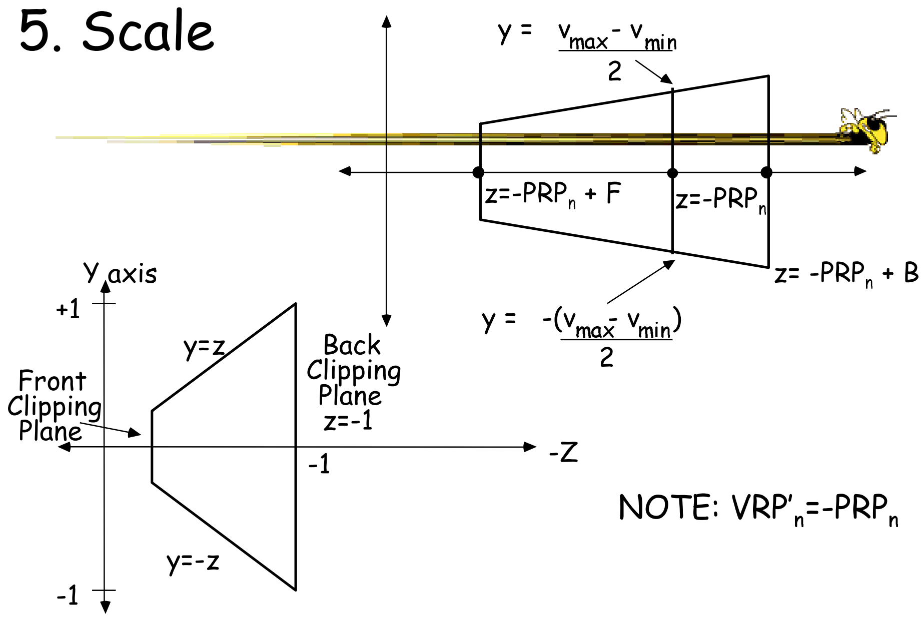
We want SH*DoP = $(0,0,DoP_z)$

$$SH = \begin{pmatrix} 1 & 0 & SHx & 0 \\ 0 & 1 & SHy & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

SHx =

SHy =

5. Scale



NOTE: $VRP'_n=-PRP_n$

5. Scale (cont.)

Scale is done in two steps:

1. First scale in x and y

   xscale = 2 PRP_n/(umax - umin)

   yscale = 2 PRP_n/(vmax - vmin)

2. Scale everything uniformly such that the back clipping plane becomes z =-1

   xscale = 1 / ($-PRP_n$ + B)

   yscale = 1 / ($-PRP_n$ + B)

   zscale = 1 / ($-PRP_n$ + B)

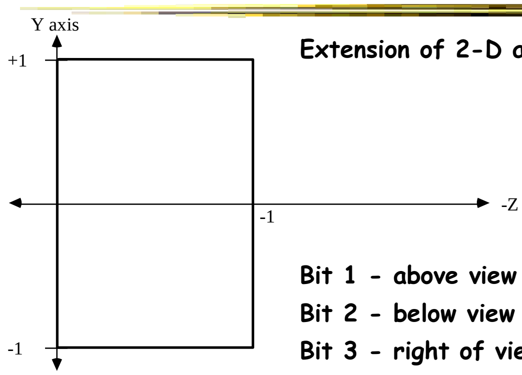
Total Composite Transformation

$$N_{per} = S_{per}\ SH_{per}\ T(-PRP)\ R\ T(-VRP)$$

Use this to transform from the viewing to the world space, then project onto the viewplane.

3-D Clipping

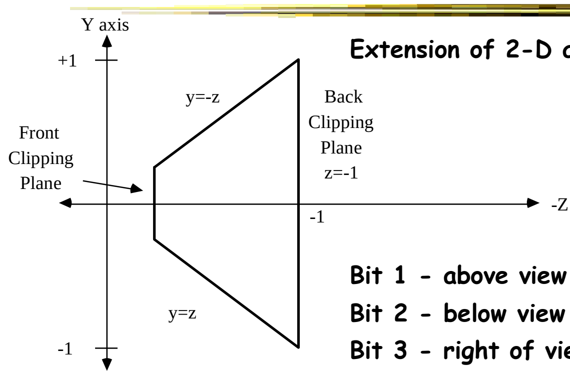
Canonical Parallel View Volume



Extension of 2-D algorithm, 6-bit outcode

Bit 1 - above view volume	$y > 1$
Bit 2 - below view volume	$y < -1$
Bit 3 - right of view volume	$x > 1$
Bit 4 - left of view volume	$x < -1$
Bit 5 - behind view volume	$z < -1$
Bit 6 - front of view volume	$z > 0$

Canonical Perspective View Volume



Extension of 2-D algorithm, 6-bit outcode

Bit 1	**- above view volume**	$y > -z$
Bit 2	**- below view volume**	$y < z$
Bit 3	**- right of view volume**	$x > -z$
Bit 4	**- left of view volume**	$x < z$
Bit 5	**- behind view volume**	$z < -1$
Bit 6	**- front of view volume**	$z > zmin$

Canonical View Volume

- Trivially accept
  - Both endpoints have a code of all zeros
- Trivially rejected
  - logical AND of the codes is not all zeros.
- Otherwise Calculate intersections.

Intersection Calculation

On the y = z plane

From parametric equation of the line:

$y0 + t(y1 - y0) = z0 + t(z1 + z0)$

Solve for t

$t = (z0 - y0) / ((y1 - y0) - (z1 - z0))$

Calculate x and y

Already know z = y

Clipping in Homogeneous Coordinates

- Two reasons:
  - Efficiency
  - Correctness

Transform Canonical Persp. Volume to Canonical Parallel Volume

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{1}{1+zmin} & \frac{-zmin}{1+zmin} \\ 0 & 0 & -1 & 0 \end{pmatrix}$$