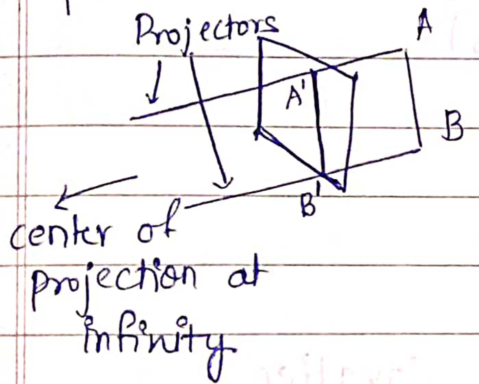
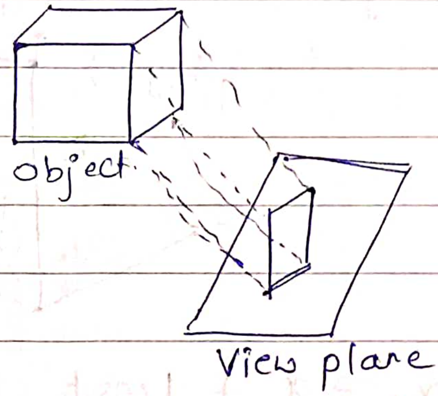
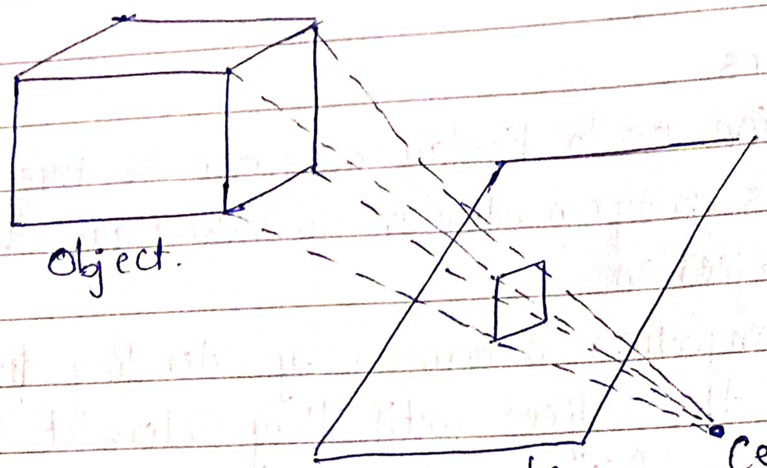
1. Parallel Projections.
- Parallel Projection use to display picture in its true shape and size.
- When projectors are perpendicular to view plane then it is called orthographic projection.
- The parallel projection is formed by extending parallel lines from each vertex on the object until they intersect the plane of the screen. The point of intersection is the projection of vertex.
- Parallel projections are used by architects and engineers for creating working drawing of the object, for complete representations require two or more views of an object using different planes.





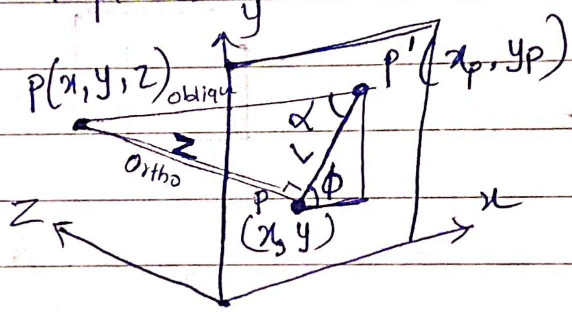
Types of Parallel ~~projectoio~~ projections:
1. Orthographic
2. Oblique.

Perspective Projections
- In Perspective Projection the center of projection is at finite distance from projection plane. This projection produces realistic views but does not preserve relative proportions of an object dimensions.
- Projections of distant object are ~~similar~~ smaller than projections of objects of same size that are closer to projection plane.
- The perspective projection can be easily described by following figure.



Oblique projection

- In oblique projection, the dirn of projection is not normal to projection of plane.

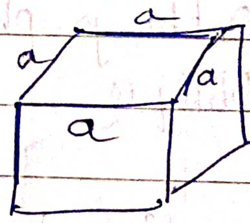


$$x_p = x + L\cos\phi$$
$$y_p = y + L\sin\phi$$

Cavalier

$\alpha = 45$

$z = L$

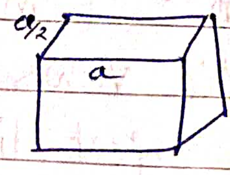


$$\tan\alpha = \frac{z}{L}$$

$$L = \frac{z}{\tan\alpha}$$
$$= zL_1$$

Cabinet

$\alpha = 63.4°$



$$\therefore x_p = x + zL_1\cos\phi \quad [\because L_1 = 1/\tan\alpha]$$

$$y_p = y + zL_1\sin\phi$$

Matrix Representation.

$$\begin{bmatrix} x_p \\ y_p \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & L_1\cos\phi & 0 \\ 0 & 1 & L_1\sin\phi & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

2] liang - Barsky algo :

1. Read two end points of line $P_1(x_1, y_1)$ & $P_2(x_2, y_2)$
2. Read two corner vertices, left top and right bottom of window (x_{wmin}, y_{max}) & (x_{wmax}, y_{wmin})
3. Calculate values of parameters P_K & q_K for $K = 1, 2, 3, 4$ such that.

$P_1 = -\Delta x$ $\quad q_1 = x_1 - \cancel{x_{wmin}}\ x_{wmin}$

$P_2 = \Delta x$ $\quad q_2 = x_{wmax} - x_1$

$P_3 = -\Delta y$ $\quad q_3 = y_1 - y_{wmin}$

$P_4 = \Delta y$ $\quad q_4 = y_{wmax} - y_1$

4. If $P_K = 0$ for any value of $k = 1, 2, 3, 4$ then.
   Line is parallel to k^{th} boundary
   If corresponding $q_K < 0$ then
   Line is completely outside the boundary.
   Therefore discard line segment & go to step 8.
   otherwise
   Check line is horizontal or vertical and accordingly check line and points with corresponding boundaries.
   If line endpoints lie within the bounded area then use them to draw line.
   otherwise
   use boundary coordinates to draw line
   And go to step 8.

5. For $K = 1, 2, 3, 4$ calculate r_K for non-zero value of P_K & q_K as follows

$$r_K = \frac{q_K}{P_K} \quad \text{for } K = 1, 2, 3, 4$$

6. Find u_1 & u_2 as given below

$$u_1 = \max \{0, r_K \mid \text{where } K \text{ takes all values for which } P_K < 0\}$$

$$u_2 = \min \{1, r_K \mid \text{where } K \text{ takes all values for which } P_K > 0\}$$

7. If $u_1 \leq u_2$ then

calculate endpoint of ~~di~~ clipped line

$$x_1' = x_1 + u_1 \Delta x$$
$$y_1' = y_1 + u_1 \Delta y$$
$$x_2' = x_1 + u_2 \Delta x$$
$$y_2' = y_1 + u_2 \Delta y$$

Draw line (x_1', y_1', x_2', y_2')

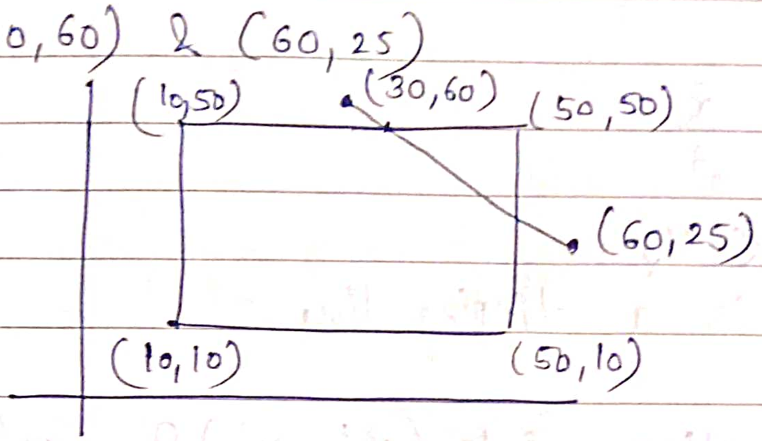
8. Stop.

Advantages

1. More efficient.
2. Only requires one division to update u_1 & u_2.
3. Window intersections of line are calculated just once.

Window Coordinates $(x_{min}, y_{min}) = (10, 10)$
$(x_{max}, y_{max}) = (50, 50)$

line $(30, 60)$ & $(60, 25)$



$$\Delta x = 60 - 30 = 30$$
$$\Delta y = 25 - 60 = -35$$

$P_1 = -\Delta x = -30$
$P_2 = \Delta x = 30$
$P_3 = -\Delta y = 35$
$P_4 = +\Delta y = -35$

$q_1 = x_1 - x_{wmin} = 30 - 10 = 20$
$q_2 = x_{wmax} - x_1 = 50 - 30 = 20$
$q_3 = y_1 - y_{wmin} = 60 - 10 = 50$
$q_4 = y_{wmax} - y_1 = 50 - 60 = -10$

$$r_1 = \frac{q_1}{P_1} = \frac{20}{-30} = \cancel{\frac{2}{3}} - \frac{2}{3}$$

$$r_2 = \frac{q_2}{P_2} = \frac{20}{30} = \frac{2}{3}$$

$$r_3 = \frac{q_3}{P_3} = \frac{50}{35} = \frac{10}{7}$$

$$r_4 = \frac{q_4}{P_4} = \frac{-10}{-35} = \frac{2}{7}.$$

$u_1 = \max(0, r_K$ / where K takes all values for which $P_K < 0$)

$$= \max\left(0, \frac{-2}{3}, \frac{\cancel{10}\,2}{7}\right)$$

$$= \cancel{\frac{10}{7}}\ \frac{2}{7}$$

$u_2 = \min(1, r_k \mid$ where k takes all values for which p_k

$= \min\left(1, \frac{2}{3}, \frac{10}{7}\right)$

$= \cancel{\frac{10}{7}}\ \frac{2}{3}$

$\because u_1 < u_2$

$\therefore$ It is a clipping line.

Let Intersection points (x_1', y_1') & (x_2', y_2')

$x_1' = x_1 + u_1(\Delta x)$

$= 30 + \frac{2}{7} \times 30$

$= 30 + \frac{60}{7}$

$= 38.57$

$y_1' = y_1 + u_1 \Delta y$

$= 60 + \frac{2}{\cancel{7}} \times (-\cancel{35}^{5})$

$= 50$

$(x_1', y') = (38.57, 50)$

$(x_2', y_2' = (50, 36.67)$

$x_2' = x_1 + u_2 \Delta x$

$= 30 + \frac{2}{\cancel{3}} \times \cancel{30}^{10}$

$= 50$

$y_2' = y_1 + u_2 \Delta y$

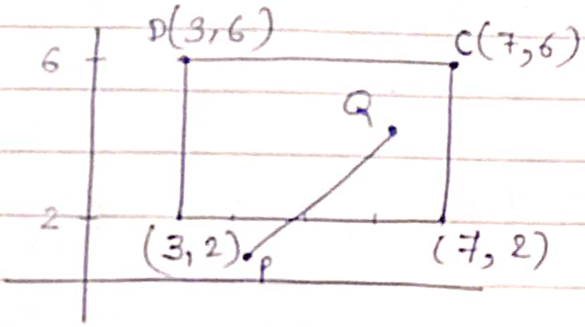
$= 60 + \frac{2}{3} \times (-35)$

$= 36.67$

3. $PQ \rightarrow P(4,1)\ Q(6,4)$

Window coordinates $\rightarrow A(3,2), B(7,2), D(3,6), C(7,6)$

$x_{wmin} = 3 \quad y_{wmin} = 2 \quad x_{wmax} = 7 \quad y_{wmax} = 6.$



$\Delta x = x_2 - x_1 = 6 - 4 = 2$

$\Delta y = y_2 - y_1 = 4 - 1 = 3.$

$P_1 = -\Delta x = -2 \qquad q_1 = x_1 - x_{wmin} = 4 - 3 = 1$

$P_2 = \Delta x = 2 \qquad q_2 = \cancel{x_{wmin}}\ x_{wmax} - x_1 = 7 - 4 = 3$

$P_3 = -\Delta y = -3 \qquad q_3 = y_1 - y_{wmin} = 1 - 2 = -1$

$P_4 = \Delta y = 3 \qquad q_4 = y_{wmax} - y_1 = 6 - 1 = 5$

$r_1 = \frac{q_1}{P_1} = \frac{1}{-2} = \frac{-1}{2}$

$r_2 = \frac{q_2}{P_2} = \frac{3}{2}$

$r_3 = \frac{q_3}{P_3} = \frac{1}{3}$

$r_4 = \frac{q_4}{P_4} = \frac{5}{3}$

$u_1 = \max\left(0, \frac{-1}{2}, \frac{1}{3}\right) = \frac{1}{3} \qquad u_2 = \min\left(1, \frac{3}{2}, \frac{5}{3}\right) = 1$

$u_1 < u_2$

It is a clipping line

$$x_1' = x_1 + u_1 \Delta x$$
$$= 4 + \frac{1}{3} \times 2$$
$$= 4 + \frac{2}{3}$$
$$= 4.67.$$

$$y_2' = y_1 + u_1 \Delta y$$
$$= 1 + \frac{1}{3} \times 5$$
$$= 2.$$

$$x_2' = x_1 + u_2 \Delta x$$
$$= 4 + 1(2)$$
$$= 6$$

$$y_2' = y_1 + u_2 \Delta y$$
$$= 1 + 1(3)$$
$$= 4.$$

$P'(4.67, 2) = (5, 2)$

$Q'(6, 4) = (6, 4)$.

4] • The method of selecting and enlarging a portion of a drawing is called windowing.

• The area ~~of~~ chosen for this display is called a window. The window is selected by world-coordinate.

• An area on a display device to which a window is mapped is viewport.

• Matrix Representation
Window to viewport

1.) Translate the lower-left of window to origin

$$T = \begin{bmatrix} 1 & 0 & -x_{wmin} \\ 0 & 1 & -y_{win} \\ 0 & 0 & 1 \end{bmatrix}$$

2.] Apply scaling to map the scene to viewport

$$S = \begin{bmatrix} \frac{xv_{max} - xv_{min}}{xw_{max} - xw_{min}} & 0 & 0 \\ 0 & \frac{yv_{max} - yv_{min}}{yw_{max} - yw_{min}} & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

3.] Inverse translation in Viewport

$$T^{-1} = \begin{bmatrix} 1 & 0 & xv_{min} \\ 0 & 1 & yv_{min} \\ 0 & 0 & 1 \end{bmatrix}$$

The composite transformation for the window to viewport transformation is given as,

$$M = T^{-1} . S . T$$

$$= \begin{bmatrix} 1 & 0 & xv_{min} \\ 0 & 1 & yv_{min} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \frac{xv_{max} - xv_{min}}{xw_{max} - xw_{min}} & 0 & 0 \\ 0 & \frac{yv_{max} - yv_{min}}{yw_{max} - yw_{min}} & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & -xw \\ 0 & 1 & -yw \\ 0 & 0 & 1 \end{bmatrix}$$

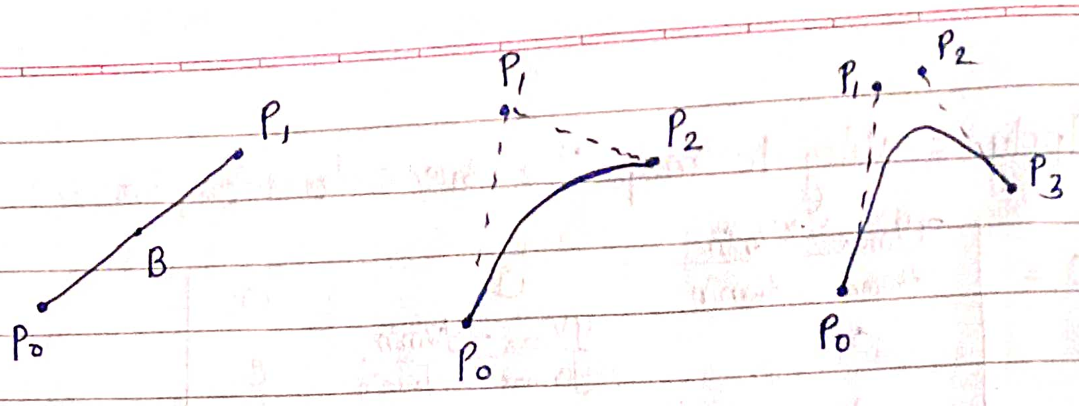
5.] Bezier Curves

- Bezier curve section can be fitted to any number of control points.
- Number of control points and their relative position gives degree of Bezier polynomials.
- With the interpolation spline Bezier curve can be specified with boundary condition or blending function.
- Consider $(n+1)$ control points position will be from P_0 to P_n where $P_i = (x_i, y_i, z_i)$
- This is blended to give position vector $p(u)$ which gives path of approximate Beizer curve as

$$p(u) = \sum_{i=0}^{n} P_i \, BEZ_{i,n} \qquad 0 \le u \le 1$$

where $BEZ_{i,n} = C(n, i) \, u^i (1-u)^{n-i}$

$$C(n, i) = \frac{n!}{i!(n-i)!}$$



- Consider $n = 3$

$$P(u) = P_i B_{i,n}(u)$$

$$= \sum_{i=0}^{3} = P_0 B_{0,3}(u) + P_1 B_{1,3}(u) + P_2 B_{2,3}(u) + P_3 B_{3,3}(u)$$

where $B_{0,3}(u) = {}^3C_0\, u^0 (1-u)^3$

$$= \frac{3!}{0!(3-0)!} \times 1(1-u)^3$$

$$= (1-u)^3$$

$$B_{1,3}(u) = 3u(1-u)^2$$
$$B_{2,3}(u) = 3u^2(1-u)$$
$$B_{3,3}(u) = u^3$$

$$P(u) = P_0(1-u)^3 + P_1\, 3u(1-u^2) + P_2\left(3u^2(1-u)\right) + BP_3 u^3$$
$$x(u) = (1-u)^3 x_0 + 3u(1-u^2)\, x_1 + 3u^2(1-u)\, x_2 + u^3 x_3$$
$$y(u) = (1-u)^2 y_0 + 3u(1-u^2)\, y_1 + 3u^2(1-u)\, y_2 + u^3 y_3$$
$$z(u) = (1-u^2)\, z_0 + 3u(1-u^2)\, z_1 + 3u^2(1-u)\, z_2 + u^3 z_3.$$

Properties of Beizer curves

- It always ~~passes~~ interpolates first and last control points and approximate the remaining two.
- The slope of derivative at beginning is along line joining the first two points and slope of derivative at the end is along the line joining the last two points.
- The degree of curve is 1 less than the no. of control points.

- Beizer curve always satisfies convex hull property.
- Beizer curves do not have local control, repositioning one control point changes the entire curve.
- Beizer curve can fit any number of control points.
- Reversing the order of control points yields the same beizer curve
- The curve begins at P_0 & ends at P_n this is so called endpoint interpolation property
- The curve is a straight line if and only if all control points are collinear.

8.] Sutherland - Hodgeman algo:

- Polygons can be clipped against each edge of window one at a time.
- Edge intersections if any are easy to find since the x or y coordinates are already known.
- Vertices which are kept after clipping against one window edge are saved for clipping against the remaining edges.
- There are different cases to find the vertices of clipped polygon:

Case 1:

Both the vertices are inside: Only the second vertex is added to output list.

Case 2:

First vertex is outside while second one is inside:

Both the point of intersection of edge with clip boundary and second vertex are added to output list.

Case 3:

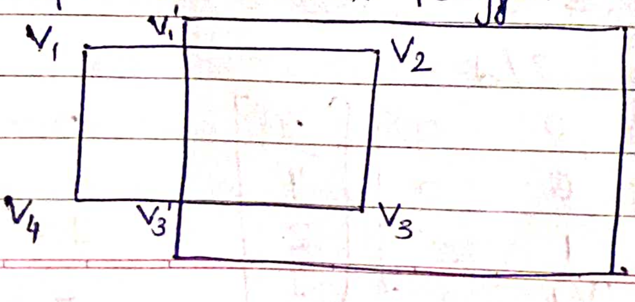
First vertex is inside while second one is outside:

Only the point of intersection of edge with the clip boundary is added to output list.

Case 4:

Both vertices are outside: No vertices are added to output list.

- Example for Convex Polygon



Output list

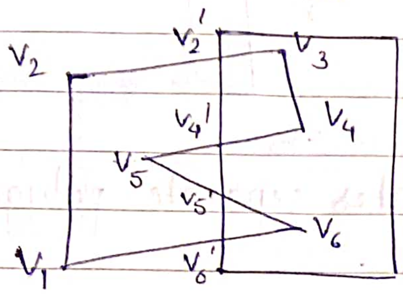
$V_1 V_2 \rightarrow V_1' V_2$

$V_2 V_3 \rightarrow V_3$

$V_3 V_4 \rightarrow V_3'$

$V_4 V_1 \rightarrow$ nothing.

• Example for Concave Polygon



Output list :

$V_1 V_2 \rightarrow$ nothing

$V_2 V_3 \rightarrow V_2' V_3$

$V_3 V_4 \rightarrow V_4$

$V_4 V_5 \rightarrow V_4'$

$V_5 V_6 \rightarrow V_5' V_6$

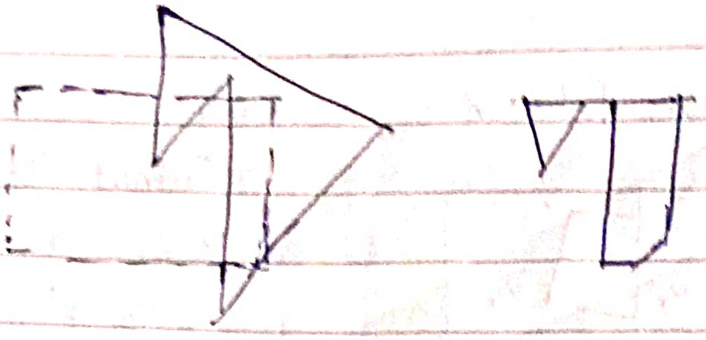
$V_6 V_1 \rightarrow V_6'$

Algo :

1. Clip a polygon by processing the polygon boundary as a whole against each window edge.
2. Beginning with the initial set of polygon vertices and traversing
~~3.~~ each polygon boundary one by one.
3. There are four possible cases when processing vertices in sequence around the ~~window~~ perimeter of polygon.
4. If first vertex is outside and second vertex is inside. Then, both the intersection point of polygon edge with window boundary and second vertex are added to output list.
5. If both vertices are inside the window boundary. Then only second vertex is added to output vertex list.
6. If first vertex is inside and second vertex is outside. Then only the edge intersection with window boundary is added to output list.
7. If both input vertices are outside the window boundary. Then, nothing is added to output vertex list.

Problem with Sutherland - Hodgeman

• Concavities can end up linked



Weiler - Atherton creates separate polygon in cases like this.

10.] Control Points:

→ In computer-aided ~~design~~ geometric design a control point is a member of a set of points used to determine the shape of a spline curve or more generally a surface or higher dimensional object.

Degree of continuity

→ The degree of continuity ~~is~~ defining the curve segment is one less that the number of defining polygon point. If there are 4 control points then degree is 3.

~~Global~~ Local control

→ ~~Global~~ Local control means moving a control point alters the shape of whole curve.

Global control

→ Global control means moving a control point does not alter or change the shape of whole curve.

11.] Cohen Sutherland line clipping algo.:

Step 1: Assign region code to both endpoint of a line depending on the position where the line endpoint is located.

Step 2:
If both endpoints region code is '0000'
→ then line is completely inside
otherwise

Step 2: Perform Bitwise OR = 0000
Accept line & draw

Step 3: Perform Bitwise AND
If result ≠ 0000 then reject the line.
else
clip the line
→ Select the endpoints
→ Find the intersection point at the window
→ Replace endpoints with the intersection pt & update the region code.

ABCD → window with A(20, 20) B(90, 20) C(90, 70) & D(20, 70)

$P_1 P_2 \rightarrow P_1(10, 30)$ $P_2(80, 90)$

$P_3 P_4 \rightarrow P_3(10, 10)$ & $P_4(70, 60)$