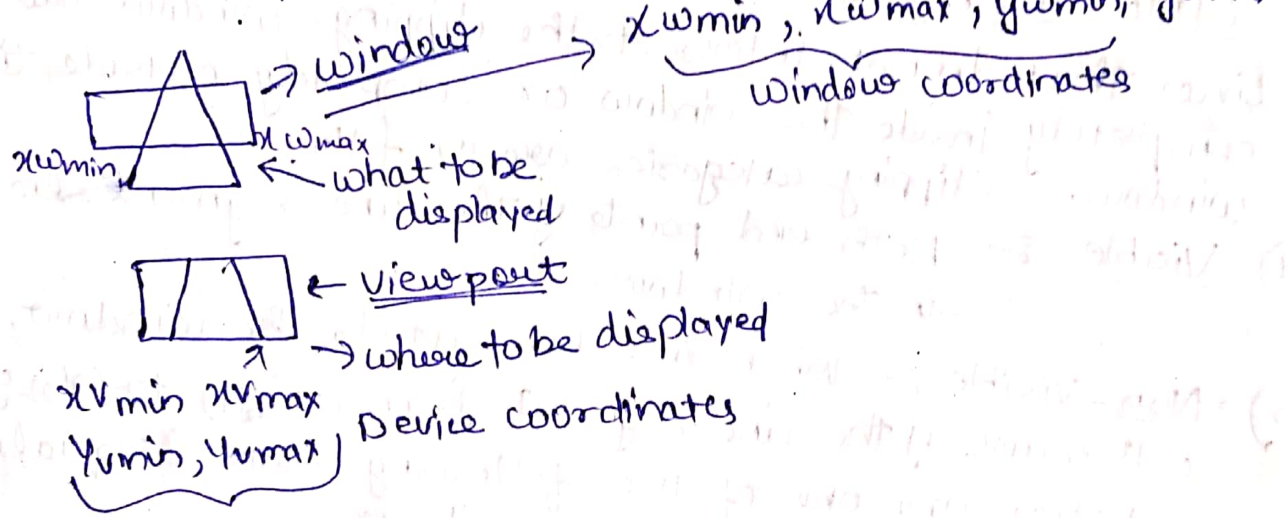
Viewing Transformation

The viewing transformation is the operation that maps a perspective view of an object in world coordinates into a physical ~~deve~~ device's display space.

To perform a viewing transformation, we deal with a finite region in the world coordinate system (WCS) called a Window. The window can be directly maped on the display area of the device called a 'View port'.



CLIPPING

In Computer Graphics, our screen act as a 2-D coordinate system. It is not necessary that each and every point can be viewed on our computer screen. We can view points, which lie in particular range (0,0) and (Xmax, Ymax).

So, Clipping is a procedure that identifies those portions of a picture that are either inside or outside of our viewing pane (computer screen).

→ POINT CLIPPING

Assuming that the clip window is a rectangle in standard position, we save a point $P(x,y)$ for display if the following inequalities are satisfied.

$$\boxed{\begin{array}{c} xw_{min} \leq x \leq xw_{max} \\ yw_{min} \leq y \leq yw_{max} \end{array}}$$

If any of these four inequalities is not satisfied, the point is clipped i.e., not saved for display.

→ LINE CLIPPING

Lines that do not intersect the clipping window are either completely inside the window or completely outside the window. Clipping categories are as follows:-

a) Visible :- Both end points of the line segment lie within the window.

b) Non-Visible :- When line lies outside the window. This will occur if the line segment from (x_1, y_1) to (x_2, y_2) satisfies any one of the following four inequalities

$x_1, x_2 > x_{max}$ $y_1, y_2 > y_{max}$

$x_1, x_2 < x_{min}$ $y_1, y_2 < y_{min}$

c) Partially visible (or clipping candidate) :- A line is partially visible when a part of its lies within the window.

Line clipping algorithms include :-

a) Cohen-Sutherland method

b) Liang-Barsky method

c) Nicholl-Lee-Nicholl method.

→ COHEN- SUTHERLAND ALGORITHM

The algorithm divides a 2D space into 9 parts, using the infinite extensions of the four linear boundaries of the window.

		Top		
	1001	1000	1010	
Left	0001	0000	0010	Right
	0101	0100	0110	
		Bottom		

Assign a bit pattern to each region as shown :-

The numbers in the figure above $\boxed{TBRL}$ are called out codes.

The bits in the outcode represents : Top, Bottom, Right, Left.

The bits in the code is set either a 1 (true) or a 0 (false)

Each bit in the code is

Fourth bit	Third bit	Second bit	First bit
1	0	1	0
T	B	R	L

→ The cohen- sutherland algorithm uses a divide and conquer strategy.

→ The line segment's endpoints are tested to see if the line can be trivially accepted or rejected.

- If the line cannot be trivially accepted or rejected, an intersection of the line with a window edge is determined and the trivial reject/accept test is repeated. This process is continued until the line is accepted.

1. Given a line segment with endpoints $P_1 = (x_1, y_1)$ & $P_2 (x_2 \; y_2)$

2. Compute 4-bit codes for each endpoint.
   - If both codes are 0000, (bitwise OR of the codes yield 000) line lies completely inside the window.

3. - If both codes have a 1 in the same bit ~~patter~~ position (bitwise AND of the codes is not 0000), the line lies outside the window, trivially rejected.

3. If a line cannot be trivially accepted or rejected at least one of the two endpoints must lie outside the window and the line segment crosses a window edge. This line must be clipped at the window edge before being passed to the drawing routine.

4. Examine one of the endpoints, say $P_1 = (x_1, y_1)$. Read P_1's 4-bit code in order TBRL

5. When a set bit (1) is found, compute the intersection I of the corresponding window edge with the line from P_1 to P_2. Replace P_1 with I and repeat the algorithm.

Intersection point are found by solving the eq^n representing the line segment and the boundary line.

- for left window
$$y = m(x_L - x_1) + y_1$$
- for right window
$$y = m(x_R - x_1) + y_1$$
- for top window
$$x = \frac{1}{m}(y_T - y_1) + x_1$$
- for bottom window
$$x = \frac{1}{m}(y_B - y_1) + x_1$$

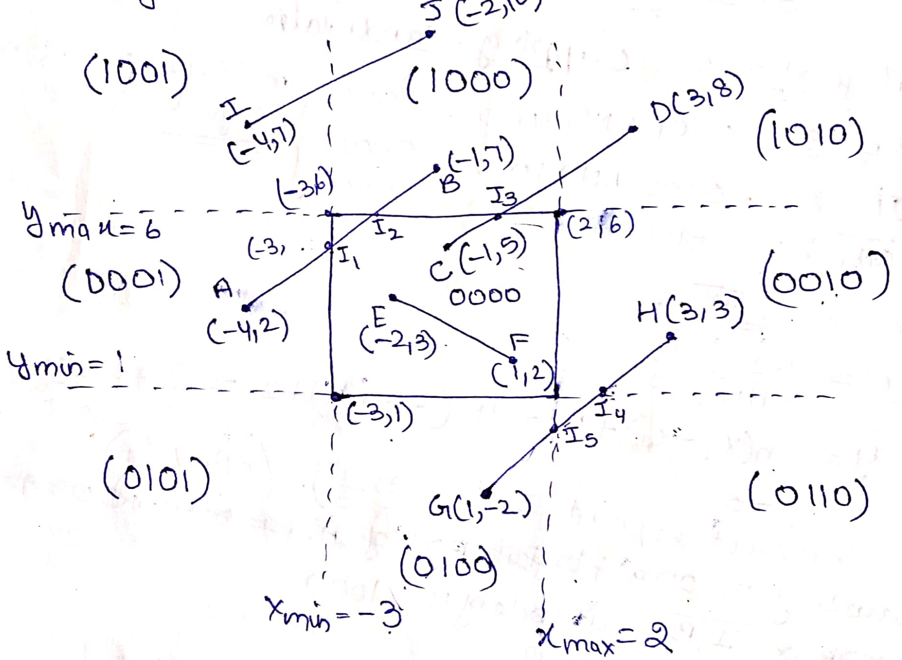
acceptable value of x & y are

$$\boxed{\begin{array}{l} x_L \le x \le x_R \\ y_B \le y \le y_T \end{array}}$$

Example.

Q. Let R be the rectangular window whose left-lower hand corner is at L(−3,1) and upper right hand corner is at R(2,6).

a) find the region codes for the endpoints in the fig.
b) find the clipping categories for the line segments.
c) Use cohen-sutherland algorithm to clip the line segments.



a) Region codes

A (−4,2) = 0001	G (1,−2) = 0100
B (−1,7) = 1000	H (3,3) = 0010
C (−1,5) = 0000	I (−4,7) = 1001
D (3,8) = 1010	J (−2,10) = 1000
E (−2,3) = 0000	
F (1,2) = 0000	

(b) **Categories**.

Category 1 :- (Visible) $\overline{EF}$ line segment is visible, because both endpoints are 0000.

Category 2 :- (Not visible) $\overline{IJ}$ line segment because having 1 at the same position & AND operation is not zero.

$$\begin{array}{l} 1001 \\ 1000 \\ \hline 1000 \text{ (not 0000)} \end{array}$$

It can trivially rejected.

Category 3 :- clipping candidates

$\overline{AB}$:- 0001 AND 1000 = 0000

$\overline{CD}$:- 0000 AND 1010 = 0000

$\overline{GH}$:- 0100 AND 0010 = 0000

neither accept nor reject.

(c) For line $\overline{AB}$:-

$m = \frac{7-2}{-1+4} = \frac{5}{3}$

$y = m(x_L - x_1) + y_1 = \frac{5}{3}(-3+4)+2 = \frac{5}{3}+2 = \frac{11}{3}$

$x = -3$

Intersection point I_1 is $(-3, \frac{11}{3})$. (clip AI_1)

outcode of I_1 (0000) ; ~~logical AND of AI_1 = 0000 AND 0001 = 0000~~

work on $\overline{I_1B}$; out code of B = 1000

$$\begin{array}{l} 0000 \\ 1000 \\ \hline 0000 \end{array} \rightarrow \text{It cannot be rejected.}$$

~~Sol~~ I_2 coordinates

$x = \frac{1}{m}(y_T - y_1) + x_1 = \frac{3}{5}(6 - \frac{11}{3}) + (-3)$

$= \frac{3}{5}\left(\frac{18-11}{3}\right) - 3$

$= \frac{7-15}{5} = -\frac{8}{5}$

$I_2(-\frac{8}{5}, 6)$

(I_2 is clipped)

The remaining segment $\overline{I_1 I_2}$ is displayed since both endpoints lie in the window.

For $\overline{CD}$

$m = \frac{8-5}{3+1} = \frac{3}{4}$ $\qquad m = \frac{3}{4}$ & $y = 6$

$x = ?$ in I_3.

$x = \frac{1}{m}(y_T - y_1) + x_1 = \frac{4}{3}(6-5) - 1 = \frac{4}{3} - 1 = \frac{1}{3}$

$I_3(\frac{1}{3}, 6)$ and its code is 0000. $\overline{I_3D}$ is clipped and the remaining segment $\overline{CI_3}$ is displayed since its both endpoints are 0000.

For line $\overline{GH}$

$m = \frac{3+2}{3-1} = \frac{5}{2}$

I_4 coordinates :- $y = 1$

$x = x_1 + \frac{1}{m}(y_B - y_1) = 1 + \frac{2}{5}(1+2) = 1 + \frac{6}{5} = \frac{11}{5}$.

$I_4(\frac{11}{5}, 1)$ and its outcode is 0010, clip $\overline{GI_4}$ and

now $\overline{I_4H}$. $\quad I_4(0010)$ & $H(0010)$ AND is not 0000

so it is rejected.

Displayed segments are :- $\boxed{CI_3,\ I_1I_2,\ EF}$

SPLINE REPRESENTATION

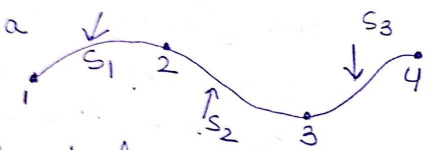
In CG, the term spline curve refers to ~~the~~ any composite curve formed with polynomial sections satisfying boundary condition at the section endpoints.

Splines can be broadly classified depending on how they fit the given set of data points, also known as the $\boxed{\text{control points}}$.

(a) Interpolated spline (b) Approximated spline

(a) Interpolated spline

Control Points :- A control point is a member of a set of points used to determine the shape of a spline curve or a surface or higher-dimensional object.

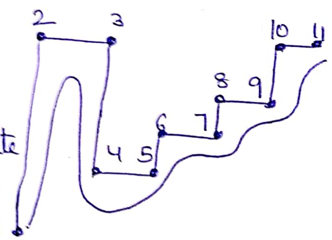


- When polynomial sections are fitted so that curve passes through each control point, the resulting curve is said to have interpolated the set of control points.

e.g. Hermite spline.

(b) Approximated spline

When polynomial are so fitted that without necessarily passing through any of the control points, it approximate the shape of the control polygon, the resulting curve is said to have approximated the set of control points.



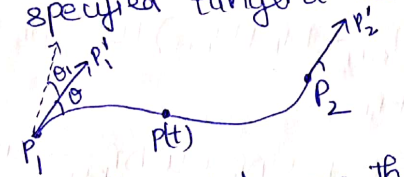
e.g. B-spline

Hermite Spline

1) It is an interpolating piecewise cubic polynomial.

$$\boxed{p(u) = au^3 + bu^2 + cu + d}$$

2) It has specified tangent at each control points

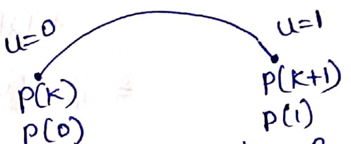


3) It has local control over the curve.

Its Mathematical behaviour :-

$$P(u) = au^3 + bu^2 + cu + d \;; \quad 0 \le u \le 1$$

$$P_x(u) = a_x u^3 + b_x u^2 + c_x u + d_x$$
$$P_y(u) = a_y u^3 + b_y u^2 + c_y u + d_y$$
$$P_z(u) = a_z u^3 + b_z u^2 + c_z u + d_z$$



$DP(0) = DP(K) =$ Derivative of P_K

$DP(1) = DP_{K+1} =$ Derivative of P_{K+1}

$$P'(u) = 3au^2 + 2bu + c$$

$P_K = P(0) = d \;;\quad P(1) = a + b + c + d = P_{K+1}$

$DP_K = P'(0) = c \;;\quad P'(1) = 3a + 2b + c = DP_{K+1}$

$$\begin{bmatrix} P_K \\ P_{K+1} \\ DP_K \\ DP_{K+1} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 \\ 3 & 2 & 1 & 0 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix}$$

$$\begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 \\ 3 & 2 & 1 & 0 \end{bmatrix}^{-1} \begin{bmatrix} P_K \\ P_{K+1} \\ DP_K \\ DP_{K+1} \end{bmatrix} = \underbrace{\begin{bmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}}_{M_H \text{ (Hermite Matrix)}} \begin{bmatrix} P_K \\ P_{K+1} \\ DP_K \\ DP_{K+1} \end{bmatrix}$$

$$P(u) = [u^3 \; u^2 \; u \; 1] \begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix} = [u^3 \; u^2 \; u \; 1] \, M_H \begin{bmatrix} P_K \\ P_{K+1} \\ DP_K \\ DP_{K+1} \end{bmatrix}$$

$$\begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix} = \begin{bmatrix} 2P_K - 2P_{K+1} + DP_K + DP_{K+1} \\ -3P_K + 3P_{K+1} - 2DP_K - 1DP_{K+1} \\ DP_K \\ P_K \end{bmatrix}$$

$$P(u) = \begin{bmatrix} u^3 & u^2 & u & 1 \end{bmatrix} \cdot \begin{bmatrix} 2P_k - 2P_{k+1} + DP_k + DP_{k+1} \\ -3P_k + 3P_{k+1} - 2DP_k - DP_{k+1} \\ DP_k \\ P_k \end{bmatrix}$$

$$P(u) = u^3(2P_k - 2P_{k+1} + DP_k + DP_{k+1}) + u^2(-3P_k + 3P_{k+1} - 2DP_k - DP_{k+1}) + uDP_k + P_k$$

$$P(u) = 2u^3P_k - 2u^3P_{k+1} + u^3DP_k + u^3DP_{k+1} - 3u^2P_k + 3u^2P_{k+1} - 2u^2DP_k - u^2DP_{k+1} + uDP_k + P_k$$

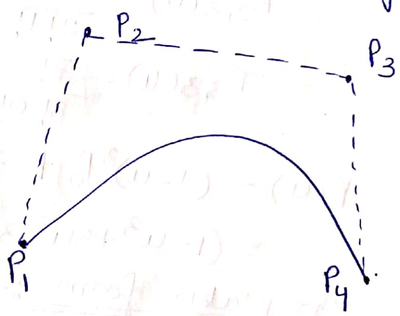
$$P(u) = P_k(2u^3 - 3u^2 + 1) + P_{k+1}(-2u^3 + 3u^2) + DP_k(u^3 + 2u^2 + u) + DP_{k+1}(u^3 - u^2)$$

$$= P_k \underline{H_0(u)} + P_{k+1} \underline{H_1(u)} + DP_k \underline{H_2(u)} + DP_{k+1} \underline{H_3(u)}$$

Hermite blending function.

CUBIC BEZIER CURVE

1. Bezier curve is a approximation spline curve i.e always passes through first & last control points.
2. Bezier curve is a parametric curve.
3. The degree of the polynomial defining the curve segment is one less than the number of defining polygon ~~segt~~ point. i.e for 4 control points, the degree of the polynomial is three.
4. The curve lies entirely within the convex hull formed by four control points.
5. Bezier curve is used in CAD, typeface, drawing etc.
6. It is easy to implement.



A Bezier spline curve can be fitted to any number of control points.

Given a set of $(n+1)$ control points $P_0, P_1 --- P_n$

$$P(u) = \sum_{i=0}^{n} P_i B_{i,n}(u) \quad ; 0 \leq u \leq 1$$

where $B_{i,n}(u)$ are the Bezier blending functions, also known as the Bernstein ~~Bezier~~ Basis functions defined as:-

$$B_{i,n}(u) = C(n,i) u^i (1-u)^{n-i}$$

$C(n,i)$ is the binomial coefficient.

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

$$P(u) = P_0 B_{0,n}(u) + P_1 B_{1,n}(u) + --- P_{n-1} B_{n-1,n}(u) + P_n B_{n,n}(u)$$

$$= P_0\{C(n,0) u^0 (1-u)^n\} + P_1\{C(n,1) u^1 (1-u)^{n-1}\} + ----$$

$$+ P_{n-1}\{C(n,n-1) u^{n-1} (1-u)\} + P_n (C(n,n) u^n (1-u)^0\}$$

For Bezier curve of degree 3 ($n=3$), we find that four ($n+1$) control points are required to specify a cubic Bezier curve segment.

$$P(u) = \sum_{i=0}^{3} P_i \, B_{i,3}(u) \qquad 0 \le u \le 1$$

$$P(u) = P_0 B_{0,3}(u) + P_1 B_{1,3}(u) + P_2 B_{2,3}(u) + P_3 B_{3,3}(u)$$

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

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

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

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

$$P(u) = (1-u)^3 P_0 + 3u(1-u)^2 P_1 + 3u^2(1-u) P_2 + u^3 P_3$$

$$= (1-u^3+3u^2-3u) P_0 + (3u^3-6u^2+3u) P_1 + (3u^2-3u^3) P_2 + u^3 P_3$$

<u>In Matrix form</u> :-

$$P(u) = \begin{bmatrix} u^3 & u^2 & u & 1 \end{bmatrix} \underbrace{\begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}}_{\text{Coefficient of } u^3,\, u^2,\, u \text{ \& constant value.}} \begin{bmatrix} P_0 \\ P_1 \\ P_2 \\ P_3 \end{bmatrix}$$

Q. construct the Bezier curve of order 3 and with 4 polygon vertices A(1,1) B(2,3), C(4,3) and D(6,4)

Sol $P(u) = (1-u)^3 P_1 + 3u(1-u)^2 P_2 + 3u^2(1-u)P_3 + u^3 P_4$

Take $u = \underline{0, \frac{1}{4}, \frac{1}{2}, \frac{3}{4}}, 1$

<u>u=0</u>

$P(0) = (1-0)^3 P_1 + 3\times 0(1-0)^2 P_2 + 3\times 0(1-0)P_3 + 0$

$P(0) = 1 P_1 = 1(1,1)$

$P(0) = (1,1).$

<u>u= 1/4</u>

$P(\frac{1}{4}) = (1-\frac{1}{4})^3 P_1 + 3\times\frac{1}{4}(1-\frac{1}{4})^2 P_2 + 3\times\frac{1}{16}(1-\frac{1}{4})P_3 + \frac{1}{64}P_4$

$= \frac{27}{64}P_1 + \frac{27}{64}P_2 + \frac{9}{64}P_3 + \frac{1}{64}P_4$

$P(\frac{1}{4}) = \left[\frac{27}{64}(1,1) + \frac{27}{64}(2,3) + \frac{9}{64}(4,3) + \frac{1}{64}(6,4)\right]$

$= \left[\frac{27}{64} + \frac{27}{64}\times 2 + \frac{9}{64}\times 4 + \frac{6}{64}, \frac{27}{64} + \frac{27\times 3}{64} + \frac{9\times 3}{64} + \frac{4}{64}\right]$

$= \left[\frac{27+36+6+54}{64}, \frac{27+81+27+4}{64}\right] = \left[\frac{123}{64}, \frac{139}{64}\right]$

$P(1/4) = (1.9218, 2.1718)$

$P(3/4) = (4.5156, 3.375)$

$P(1/2) = (3.125, 2.875)$

$P(1) = (6,4)$

Q. Find the eqⁿ of Bezier curve which passes through points (0,0) & (-2,1) and is controlled through points (7,5) & (2,0).

Sol. $P_1(0,0)$ $P_2=(7,5)$ $P_3(2,0)$ & $P_4(-2,1)$

P_1 ↓ Ist control point

P_4 ↓ Last control point.

$$P(u) = [u^3 \; u^2 \; u \; 1] \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} P_1 \\ P_2 \\ P_3 \\ P_4 \end{bmatrix}.$$

$$= [u^3 u^2 u 1] \begin{bmatrix} -1 & 3 & -3 & \phi \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0 & 0 \\ 7 & 5 \\ 2 & 0 \\ -2 & 1 \end{bmatrix}$$

$$= = [u^3 \; u^2 \; u \; 1] \begin{bmatrix} 13 & 16 \\ -36 & -30 \\ 21 & 15 \\ 0 & 0 \end{bmatrix}$$

$$\boxed{P(u) = (13u^3 - 36u^2 + 21u)\,(16u^3 - 30u^2 + 15u)}$$

eqⁿ of Bezier curve

CONTINUITY

Continuity conditions :- continuity conditions explains that how smoothly the adjacent pieces join each other at the common endpoints.

There are two types of continuity b/w two adjacent curves pieces :-

1) Parametric Continuity ~~conditions~~ (C)
2) Geometric continuity (G)

1) Parametric Continuity conditions :- If each section of a spline is described with a set of parametric coordinate functions of the form

$$x = x(u),\ y = y(u),\ z = z(u)$$

We set parametric continuity by matching the parametric derivative of adjoining curve sections at their common boundary.

2) Geometric Continuity :- An alternate method for joining two successive curve sections is to specify conditions for geometric continuity. In this case, we only require parametric derivatives of two sections to be proportional to each other at their common boundary instead of equal to each other.

Order of continuity :-

a) Zero order continuity

When two curves share a common endpoint without regard to the shape of the curves at the join-point.

i.e. value of $x, y \& z$ evaluated at $t=1$ for the first curve section are equal to the value of x, y and z evaluated at $t=0$ for the next curve.

This is known as zero-order parametric as well as geometric continuity. denoted as C^0 and G^0 resp.

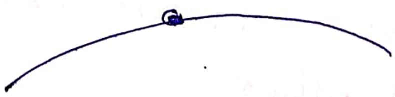
b) First Order Continuity

If the tangent vector i.e. the parametric first derivatives are equal at the segment's join-point, then the curves have first order parametric continuity or C^1 continuity.

→ slopes are equal at the join point.

c) Second Order Continuity

Both the first & second parametric derivatives of the two curve sections are equal at the joint, while second order geometric continuity or G^2 continuity requires the derivatives to be proportional to each other, if not equal.



Conditions for smoothly Joining Cubic Bezier Curve Segments.

Q. What are the major differences b/w Bezier curve, B-Spline and Hermite curves? Make a comparison b/w them and suggest which curve should be used in CG? (19)

Sol

1. Explain definition
2. Write basic expression
3. Properties
4. Basic Diagram
5. About control points
6. whether approximation & interpolated.

First, the no. of specified polygon vertices fixes the order of the resulting polynomial which defines the curve.

Second, the value of the blending function is non-zero for all parameter values over the entire curve.

Due to this change in one-vertex, changes the entire curve and this eliminates the ability to produce a local change within a curve.

⇒ The B-spline basis is non-global. It is non-global because each vertex B_i is associated with a unique basis function.

The B-spline basis allows the order of the basis function and hence the degree of the resulting curve is independent on the no. of vertices.

If $P(u)$ be the position vectors along the curve as a function of the parameter u, a B-spline curve is given by :-

$$P(u) = \sum_{K=0}^{n} P_K \, B_{K,d}(u), \quad U_{min} \leq u \leq U_{max}$$

$$2 \leq d \leq n+1$$

where P_K are an input set of $(n+1)$ control points

4

$B_{k,d}$ are polynomials of degree $(d-1)$, where parameter $\underline{d}$ can be chosen to any integer value in the range from 2 upto no. of control points, $(n+1)$:-

$$B_{k,1}(u) = \begin{cases} 1 & \text{if } u_k \le u \le u_{k+1} \\ 0 & \text{otherwise} \end{cases}$$

$$\boxed{B_{k,d}(u) = \frac{u - u_k}{u_{k+d-1} - u_k} B_{k,d-1}(u) + \frac{u_{k+d} - u}{u_{k+d} - u_{k+1}} B_{k+1,d-1}(u)}$$

where each blending function is defined over $\underline{d}$ subintervals of the total range of u.

The selected set of sub-intervals end-points u_j is referred to as a <u>knot vector</u>. We can choose any values for the subinterval endpoints satisfying the relation $u_j \le u_{j+1}$. Value for u_{min} & u_{max} then depend on the no. of control points we select, the value we choose for parameter d and how we set up the subintervals (knot vector).

$\Rightarrow$ For a cubic B-spline, the degree $\boxed{d-1=3}$

So $\boxed{d=4}$ and hence no. of control points $(n+1)$ required is atleast 4 ($\because n+1 \le d$).

$$P(u) = P_0 B_{0,4}(u) + P_1 B_{1,4}(u) + P_2 B_{2,4}(u) + P_3 B_{3,4}(u)$$

for knot vector $[0 \; 1 \; 2 \; 3 \; 4 \; 5 \; 6 \; 7]$

$\boxed{n+d}$ ~~degree~~

$\cancel{d=3}$ $d=4$ $\quad n=4 \quad = 8$

$\boxed{0-7}$

Properties of B-spline

1) The polynomial curve has degree $d-1$ and C^{d-2} continuity over the range of u.
2) For $n+1$ control points, the curve is described with $n+1$ blending functions.
3) Each section of the spline curve is influenced by d control points.
4) Any one control point can affect the shape of at most "d" curve sections.
5) The range of parameter u is divided into $(n+d)$ sub-intervals by the $n+d+1$ values specified in the knot vector.

Three types of knot vectors

a) Uniform b) Open-Uniform c) Non-Uniform

a) Uniform :- When the spacing b/w the knot values is constant, the resulting curve is called uniform B-spline. e.g. [0 1 2 3 4 5 6 7]

b) Open-Uniform :- for open, uniform B-splines, the knot spacing is uniform except at the ends where knot values are repeated d times. e.g.
[0, 0, 1, 2, 3, 3] $d=2$ & $n=3$.

c) Non-uniform B-splines :- we can choose multiple internal knot values & unequal spacing b/w knot values
e.g. [0, 1, 2, 3, 3, 4].

They provide increased flexibility in controlling a curve shape.

3-D Projections

Projection is a process of representing a three-dimensional object or scene into two-dimensional medium. i.e projection is nothing but a shadow of the object.

We can say that, projection transforms points in a coordinate system of dimension 'n' into points in a coordinate system of dimension less than 'n'.

- Centre of Projection :- The point from where projection is taken. It can either be light source or eye position.
- Projection plane :- The plane on which projection of the object is formed.
- Projectors :- Lines emerging from centre of projection and hitting the projection plane after passing through a point in the object to be ~~formed~~ projected.
  i.e projectors are lines from an arbitrary point called the centre of projection, through each point in an object.

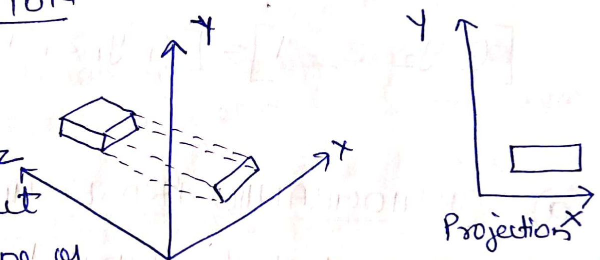
→ Projection can be divided into two basic classes :- perspective and parallel.

Scanned by CamScanner

Categories of Projection

- Projection
  - Parallel Projection (• centre of projection is at infinity • All projectors are parallel to each other)
    - Orthographic Projection
    - Axonometric Projection
      - Isometric Projection
      - Dimetric Projection
      - Trimetric Projection
    - Oblique projection
      - Cavalier Projection
      - Cabinet Projection
  - Perspective projection (• centre of projection is at finite distance • Projectors are not parallel).
    - Single Point Projection
    - Two-Point Projection
    - Three Point Projection.

PARALLEL PROJECTION



A parallel projection is formed by extending parallel lines from each vertex of the object until they intersect the plane of the screen. The point of intersection is the projection of the vertex.

- All projectors are parallel to each other
- Centre of projection is at infinity.
- Parallel projection preserves the relative properties of objects in x & y direction.

Now, suppose direction of projection is given by the vector $[x_p\ y_p\ z_p]$ & image is to be projected onto the xy plane.

Point on the object at (x_1, y_1, z_1), we need to determine where the projected point (x_2, y_2) will lie :-

$$x = x_1 + x_p u \quad \text{— (1)}$$
$$y = y_1 + y_p u \quad \text{— (2)}$$
$$z = z_1 + z_p u \quad \text{— (3)}$$

If $z = 0$; Put in (3)

$\cancel{0 = z_1 + z_1(\ldots) \Rightarrow}$

$$0 = z_1 + z_p u \Rightarrow \quad u = -\frac{z_1}{z_p} \quad \text{Put in (1) \& (2)}$$

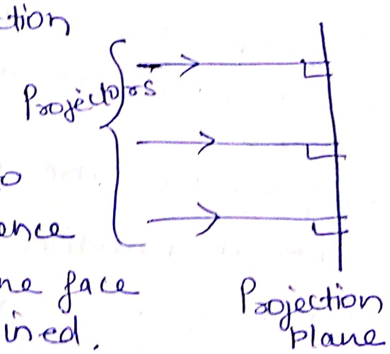
$$x_2 = x_1 - x_p \frac{z_1}{z_p} \Rightarrow x_1 - z_1\left(\frac{x_p}{z_p}\right)$$

$$y_2 = y_1 - z_1\left(\frac{y_p}{z_p}\right)$$

$$[x_2\ y_2\ z_2\ 1] = [x_1\ y_1\ z_1\ 1] \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ -\frac{x_p}{z_p} & -\frac{y_p}{z_p} & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

a) ORTHOGRAPHIC PROJECTION

It is a part of the parallel projection in which the center of projection lies at infinity. In orthographic projection, the projectors are ⊥ to each ~~othe~~ projection plane and hence the size & shape of a single plane face of an object is ~~at~~ ~~infinit~~ obtained.



- Orthographic projection do not change the length of line segments which are parallel to projection plane. Other lines are projected with reduced length.

- Used for engineering drawing. It is the projection on one of the coordinate planes i.e $x=0$ $y=0$ $z=0$

for $x=0$

$$P_x = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$P_y = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad P_z = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

b) AXONOMETRIC PROJECTIONS

Axonometric projections are orthographic projections in which the direction of projection is not parallel to any of the three principal axis.

The construction of an axonometric projection is done by using rotation and translation to manipulate the object such that atleast three adjacent faces are shown.

- An axonometric projection shows its true shape only when a face is parallel to the plane of projection.

Parallel lines are equally foreshortened.

- The forshortening factor is the ratio of the projected length of a line to its true length.

Axonometric projection is of three types :-

A) Isometric Projection :- The direction of projection make equal angles with all three principal axis.

In ~~a~~ isometric projection all three foreshortening factors are kept equal.

B) **Dimetric Projection** :- The direction of projection makes equal angles with exactly two of the principal axis.

- Two forshortening factors are equal & third arbitrary.

C) **Trimetric Projection** :- The direction of projection makes unequal angles with the three principal axis.

It is formed by arbitrary rotation in arbitrary order about any or all of the coordinate axis followed by parallel projection onto $z=0$ plane.

- All foreshortening factors are different.

☆ Foreshortening factors

Isometric $\Rightarrow$ $\boxed{f_x = f_y = f_z = \sqrt{\cos^2\theta} = \sqrt{\cos^2 45^\circ} = \sqrt{\frac{2}{3}} = \underline{\underline{0.8165}}}$

Dimetric $\Rightarrow$

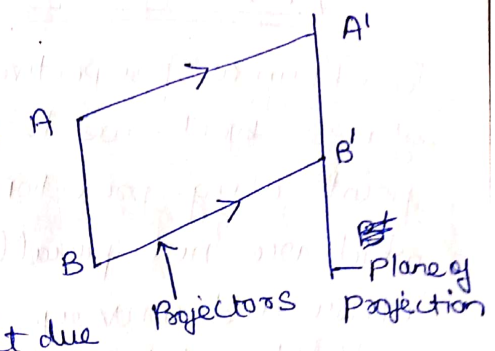
$$\boxed{\begin{aligned} f_x &= \sqrt{\cos^2\phi + \sin^2\phi \sin^2\theta} = f_z \\ f_y &= \sqrt{\cos^2\theta} = \cos\theta. \end{aligned}}$$

Trimetric $\Rightarrow$

$$\boxed{\begin{aligned} f_x &= \sqrt{(x'_x)^2 + (y'_x)^2} \\ f_y &= \sqrt{(x'_y)^2 + (y'_y)^2} \\ f_z &= \sqrt{(x'_z)^2 + (y'_z)^2} \end{aligned}}$$

c) OBLIQUE PROJECTION

When the angle b/w the projectors and the plane of projection is not equal to 90° then the projection is called oblique projection



for. e.g. Shadow of any object due to sunlight.

→ Oblique projections are of two types :-

a) Cavalier Projection b) Cabinet Projection

a) Cavalier Projection :- The cavalier projection is formed when the angle b/w the oblique projectors and the plane of projection is 45° and foreshortening factors for all three principal directions are equal i.e. $\boxed{F=1}$

- The resulting figure in cavalier projection is thicker.

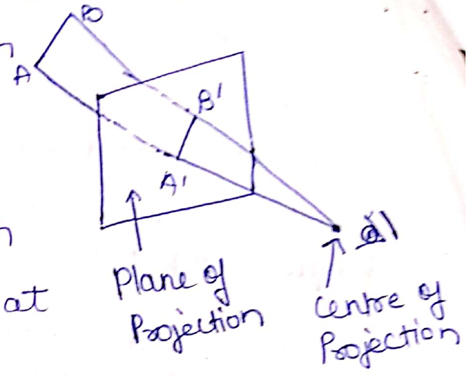
b) Cabinet Projection :- A cabinet Projection is used to correct the distortion that is produced by cavalier projection. Foreshortening factor is one-half in cabinet projection. $F=1/2$

- Angle b/w plane of projection & projectors is $\boxed{\cot^{-1}(1/2) = 63.43°}$

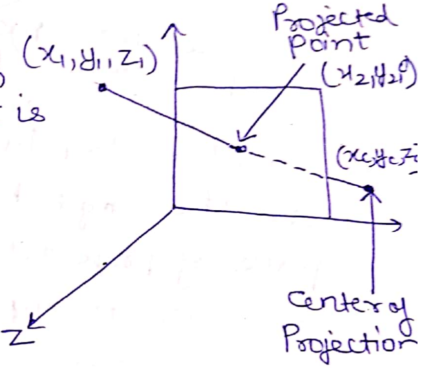
$$\begin{bmatrix} x_p \\ y_p \\ z_p \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & f\cos\theta & 0 \\ 0 & 1 & f\sin\theta & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

PERSPECTIVE PROJECTION

To obtain a perspective projection of a 3D object, we transforms points along projection lines which are not parallel to each other and converge to meet at a finite point known as the projection reference point or the center of projection.



Let us consider the center of projection is at (x_c, y_c, z_c) and point on object is $(x_1\ y_1\ z_1)$



$$\left.\begin{aligned} x_2 &= x_c + (x_1 - x_c)u \\ y_2 &= y_c + (y_1 - y_c)u \\ z_2 &= z_c + (z_1 - z_c)u \end{aligned}\right\}.$$

$$z_2 = 0$$

$$u = \frac{-z_c}{z_1 - z_c}$$

$$x_2 = x_c - \frac{z_c}{z_1 - z_c}(x_1 - x_c) = \frac{x_c z_1 - \cancel{x_c z_c} - x_1 z_c + \cancel{x_c z_c}}{z_1 - z_c}$$

$$\boxed{x_2 = \frac{x_c z_1 - x_1 z_c}{z_1 - z_c}}$$

$$\boxed{y_2 = \frac{y_c z_1 - y_1 z_c}{z_1 - z_c}}$$

Matrix can be written as:-

$$[x_2\ y_2\ z_2\ 1] = [x_1\ y_1\ z_1\ 1] \begin{bmatrix} -z_c & 0 & 0 & 0 \\ 0 & -z_c & 0 & 0 \\ x_c & y_c & 0 & 0 \\ 0 & 0 & 0 & -z_c \end{bmatrix}$$

Vanishing Points

- Projections of distant objects are smaller than the projections of objects of the same size that are close to the projection plane (or center of projection). This is known as the perspective ~~projection~~. foreshortening.

- The illusion that after projection certain set of parallel lines appear to meet at some point on the projection plane. These points are called Vanishing points.

- The vanishing point for any set of lines that are parallel to one of the principal axis of an object is referred to as a principal vanishing point or axis vanishing point

We can control the no. of principal vanishing point with the orientation of the projection plane and perspective projections are classified as.

perspective projections:
- One-point projection
- two-point projection
- three-point projection.

→ Single point perspective transformation

One point perspective projection occurs when only one principal axis intersects the plane of projection.

3 types of single point perspective transformations:-

☆ when projectors are located at x-axis :-

$$[x' \ y' \ z' \ 1] = [x \ y \ z \ 1] \begin{bmatrix} 1 & 0 & 0 & p \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$[x' \; y' \; z' \; 1] = [x \quad y \quad z \quad (px+1)] \; ; \; px+1 \neq 1.$$

$$[x' \; y' \; z' \; 1] = \left[\frac{x}{px+1} \quad \frac{y}{px+1} \quad \frac{z}{px+1} \quad 1\right]$$

Centre of projection at $[-1/p \quad 0 \quad 0 \quad 1]$ and vanishing point located on x-axis is at $[1/p \quad 0 \quad 0 \quad 1]$

Similarly for y & z.

y-axis $$[x' \; y' \; z' \; 1] = \left[\frac{x}{qy+1} \quad \frac{y}{qy+1} \quad \frac{z}{qy+1} \quad 1\right]$$

Centre of projection at $[0 \quad -1/q \quad 0 \quad 1]$ & vanishing point located on y-axis at $[0 \quad 1/q \quad 0 \quad 1]$

z-axis $$[x' \; y' \; z' \; 1] = \left[\frac{x}{rz+1} \quad \frac{y}{rz+1} \quad \frac{z}{rz+1} \quad 1\right]$$

Center of projection at $[0 \quad 0 \quad -1/r \quad 1]$ & vanishing point located on z-axis at $[0 \quad 0 \quad 1/r \quad 1]$.

→ Two Point Perspective transformation

Two point or two principal vanishing point perspec-tive projection occurs when the plane of projection intersects exactly two of the principal axis.

$$[x' \; y' \; z' \; 1] = [x \; y \; z \; 1] \begin{bmatrix} 1 & 0 & 0 & p \\ 0 & 1 & 0 & q \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$= [x \quad y \quad z \quad px+qy+1]$$

$$= \left[\frac{x}{px+qy+1} \quad \frac{y}{px+qy+1} \quad \frac{z}{px+qy+1} \quad 1\right]$$

Three point projection occurs when the projection plane intersects all three of the principal axis i.e. none of the principal axis is parallel to the projection plane.

$$[x' \ y' \ z' \ 1] = [x \ y \ z \ 1]\begin{bmatrix} 1 & 0 & 0 & p \\ 0 & 1 & 0 & q \\ 0 & 0 & 1 & r \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

$$= [x \quad y \quad z \quad (px+qy+rz+1)]$$

$$= \left[\frac{x}{(px+qy+rz+1)} \quad \frac{y}{(px+qy+rz+1)} \quad \frac{z}{px+qy+rz+1} \quad 1\right]$$

Center of Projection :-
$[-1/p \quad 0 \quad 0 \quad 1]$
$[0 \quad -1/q \quad 0 \quad 1]$
$[0 \quad 0 \quad -1/r \quad 1]$

Vanishing Point :-
$[1/p \quad 0 \quad 0 \quad 1]$
$[0 \quad 1/q \quad 0 \quad 1]$
$[0 \quad 0 \quad 1/r \quad 1]$.

Rendering is the process of producing realistic images or pictures. Producing realistic images involves both physics and psychology.

- Light i.e. electromagnetic energy, reaches the eye after interacting with the physical environment.
- The human brain interprets this information for visualization of the object.

ILLUMINATION MODELS.

We concentrate on the location and qualities of the light that falls on the objects and the way in which object interacts with it. A model for the interaction of light with a surface is called an illumination model.

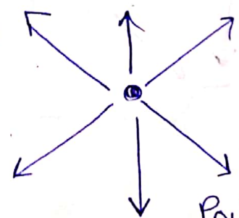
- Light sources

Light sources that illuminates an object are of two basic types :-

emitting sources :- ~~light~~ bulb & sun.

reflecting sources :- walls of a room.

→ when the surface of the object which we want to illuminate is bigger than the surface of light emitting source then we are calling that light emitting source as point source. for e.g- Sun



→ when the surface of light emitting source is greater than the surface of object then we are refering it as distributed light source. e.g. neon-tube light.

When light is incident on an opaque surface, part of it reflected and part is absorbed. The amount of incident light reflected by a surface depends on the type of material.

- Shiny materials reflects more of the incident light and dull surfaces absorb more of the incident light.

☆ Ambient Light :-

A surface that is not exposed directly to a light source still will be visible if nearby objects are illuminated.

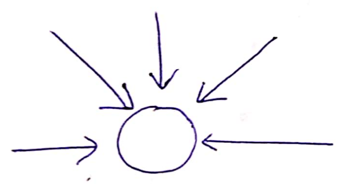
In our basic illumination model, we can set a general level of brightness for a scene. This is a simple way to model the combination of light reflections from various surfaces to produce a uniform illumination called the ambient light, or background light.

Ambient light has no spatial or directional characteristics. The amount of ambient light incident on each object is a constant for all surfaces and over all directions.

$$I = K_a I_a$$

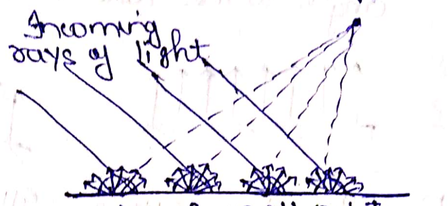
K_a → ambient reflection coefficient

I_a → incident ambient intensity



DIFFUSE ILLUMINATION

Diffuse illumination is a background light which is reflected from walls, floor & ceiling. It is uniform from all directions.



Light from all points on the surface reaches the viewer.

We will assume that there is as much light going up as there is going down and that there is the same amount going right as there is going left. When the light in all directions is of same amount then we can say that the reflection are constant over each surface of the object and they are independent of viewing direction, such a reflection is called diffuse reflection.

Diffuse Reflection

$\Rightarrow$ (R), Coefficient of reflection $= \dfrac{\text{light reflected from the surface}}{\text{Total incoming light}}$

& $0 < R < 1$ ≈ 1 for white surface, ≈ 0 for black surface

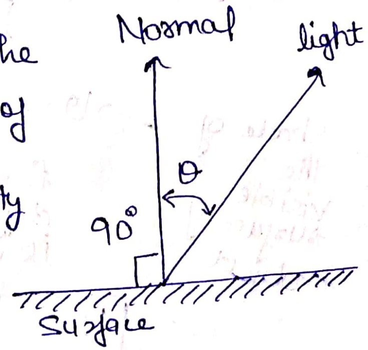
a) **Lambert's Cosine Law**

This law states that the intensity of the reflected light depends on the angle of illumination.

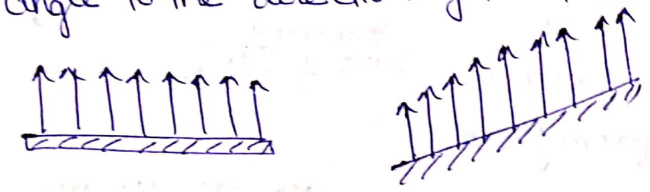
$I = BR$ → Object reflectivity

I: intensity of light coming from any visible surface

B: Intensity of background light

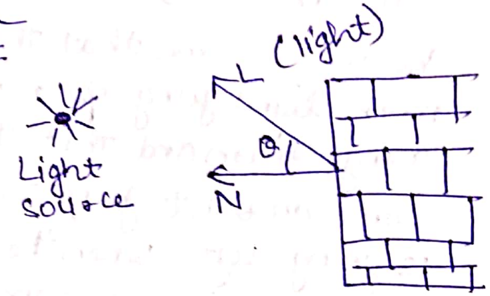


~~Accor~~.
A surface that is ⊥ to the direction of the Incident light appears brighter than a surface that is at an angle to the direction of the incoming light.



b) Point - source Illumination

Point sources are real world sources of light such as sun, bulb, candles etc. They emit rays from a single point.



- Position & orientation of the objects surface relative to the light source will determine how much light the surface receive.

Surfaces which are near the light source and facing towards light will receive more light.

- The illumination is decreased by a factor of $\cos\theta$, θ is the angle b/w the direction of Light (L) & Normal (N) to the plane.

$\boxed{\cos\theta = L \cdot N}$

Shade of the visible surface of object. $\rightarrow$ $V = BR + PR\cos\theta$

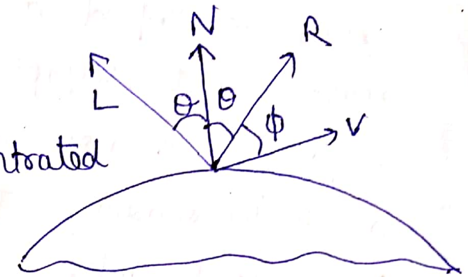
BR ← from diffuse illumination

P ← intensity comes from point source

R → diffuse reflectivity coefficient

$\boxed{V = BR + PR(N \cdot L)}$

→ Specular reflection is the result of total or near total, reflection of the incident light in a concentrated region around the specular reflection angle.



Due to specular reflection, the surface appears to be not in its original color, but white, the color of incident light.

- The specular reflection angle equals the angle of the incident light, with the two angles measured on opposite sides of the unit Normal surface vector N.

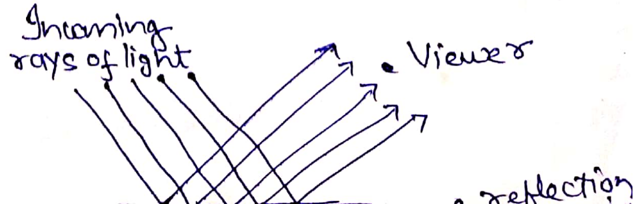
R = Unit vector in the direction of ideal specular reflection.

L = L is the unit vector directed toward the point light source

V = V is the unit vector pointing to the viewer from the surface position

The Angle φ between Vector R and vector V is called viewing angle.

☆ For an ideal reflector, (perfect mirror) incident light is reflected only in the specular reflection direction. i.e. φ = 0



→ Phong - Specular - Reflection Model.

Phong developed a popular illumination model for non-perfect reflectors. It assumes that max. specular reflection occurs when ϕ is zero and falls off sharply as ϕ increases. This rapid fall-off is approximated by $\cos^{n_s}\phi$, where $\boxed{n_s}$ is the specular reflection parameter determined by the type of surface.

The intensity of specular reflection depends on the material properties of the surface and the angle of incidence, as well as other factors such as the polarization and colour of the incident light.

Phong - specular Reflection :-

$$I_{spec} = W(\theta) I_l \cos^{n_s}\phi$$

$W(\theta)$ → set to constant k_s, material's specular reflection coefficient, $0 < k_s < 1$

I_l → Intensity of the light source.

ϕ → viewing angle relative to the specular reflection direction R.

$$\boxed{I_{spec} = k_s I_l (V \cdot R)^{n_s}}$$