Visible Surface Detection ≈ Hidden Surface Removal

Identify those surfaces which are blocked/hidden from view & removal of these surfaces.

Object Space Method (through object)

Visibility

for (each object in world)
Determine those parts of the object whose view is unobstructed by other parts of it. Draw the parts)

- Back face detection
- Painter's Algo
- Roberts Algo

Image Space Method (through pixel)

(——— pixel in image
Determine object closest to viewer that is intercepted by projector through pixel.
Draw pixel with appropriate color

- Depth buffer / Z buffer
- Scan line
- Ray tracing

(i) Object Space Method
(ii) Image Space "

Inside Outside Test

$Ax + By + Cz + D < 0$ ← Inside

Normal vector (A, B, C)

Surface — 90°

$V \cdot N > 0$ (Surface backface)

$V \cdot N < 0$ (Front face)

Left Hand Model

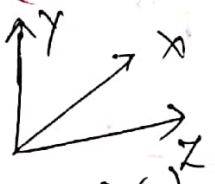
$C > 0$

$V = (0, 0, V_z)$

Parallel to z axis

$V \cdot N = V_z \cdot C$

$(0, 0, V_z)(A, B, C)$



Right Hand System

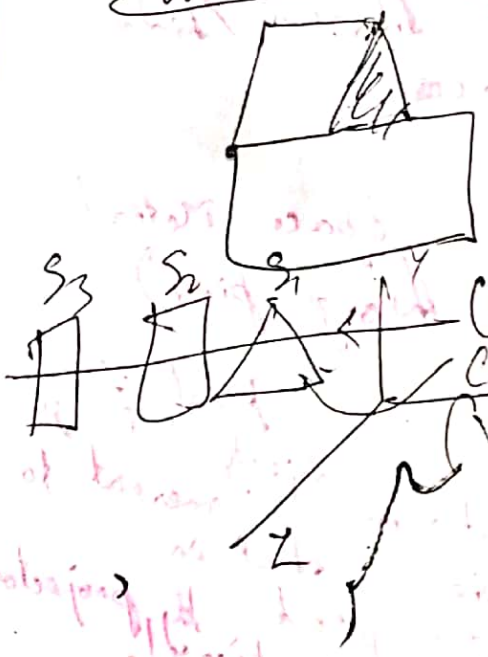
$(0, 0, V_z)$ $V_z = -V_z$

$C \leq 0 \Rightarrow -V_z \cdot (-C) = V_z C$

$V \cdot N > 0$

face → backface

Z buffer method / Depth Buffer —
Such object overlap back & front



After

Algorithm —
- Buffer (x, y)
→ Z value calculation
Refresh (x, y)
- Intensity of (x, y)

Z coordinate - Normalized
0 < Z < 1
Back Clipping plane
front Clipping plane

1) Initialize both the buffer i.e Buffer (x, y) = 0 & Refresh (x, y) = Background.

2) Calculate Z value for each position in the surface & then if Z > depth (x, y)
depth (x, y) = Z &
Refresh (x, y) = Surface (x, y)

3) After procesing all the surface we will get visible surface in depth (x, y) & intensity values in Refresh (x, y)

Eq of plane
$Ax + By + Cz + D = 0$
$z = -\frac{Ax - By - D}{C}$

$z' = -\frac{A(x+1) - By - D}{C}$

$z' = -\frac{Ax - A - By - D}{C}$

$z' = z - A/C$

Opaque —
Transparent object ✗