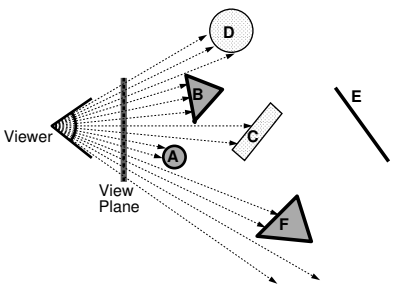
Ray Tracing

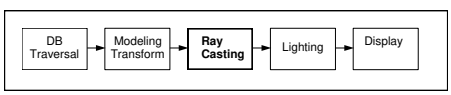
Thomas A. Funkhouser

Overview

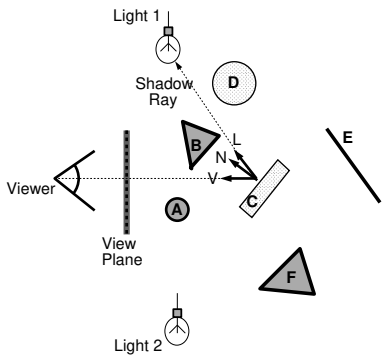
- Ray casting revisited
- Recursive ray tracing
- Ray–surface intersections
- Acceleration techniques
- Sampling

Ray Casting

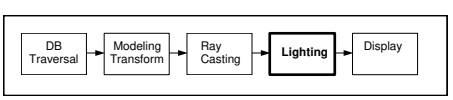




Ray Casting



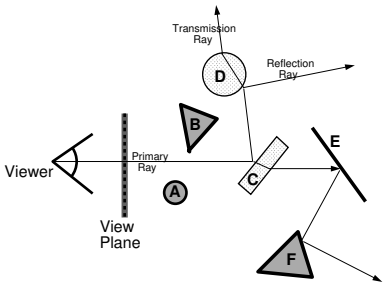
$$I = I_E + K_A I_A + \sum_{L_i} (K_D(N \bullet L_i) + K_S(V \bullet R_i)^n) S_i I_{L_i}$$

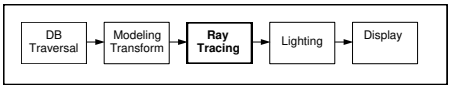


Ray Tracing

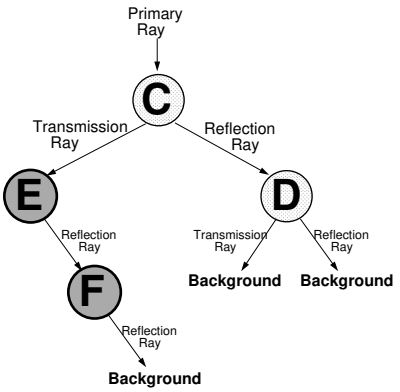
- **Goal**
  - Simulate indirect illumination

Recursive Ray Tracing

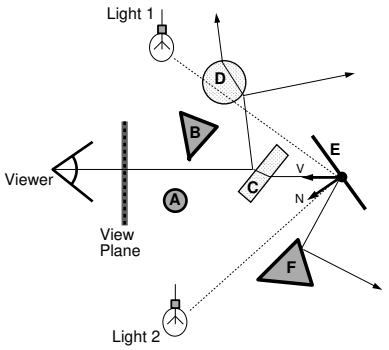




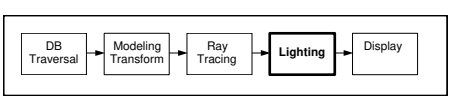
Ray Tracing Tree



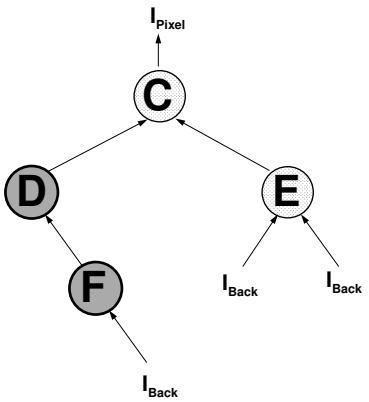
Ray Tracing Lighting



$$I = I_E + K_A I_A + \sum_{L_i} (K_D (N \bullet L_i) + K_S (V \bullet R_i)^n) S_i I_L + K_S I_R + K_T I_T$$



Ray Tracing Lighting



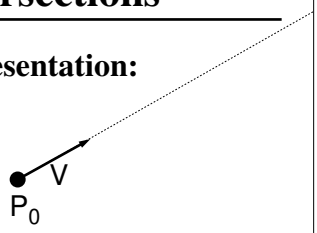
$$I=I_E+K_AI_A+\sum_{L_i}(K_D(N\bullet L_i)+K_S(V\bullet R_i)^n)S_iI_L+K_SI_R+K_TI_T$$

Ray Intersections

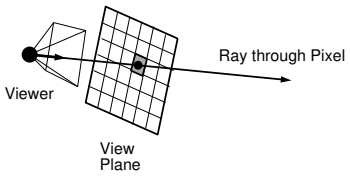
- **Ray representation:**

$P = P_0 + tV$

$t >= 0$



- **Ray through a pixel:**



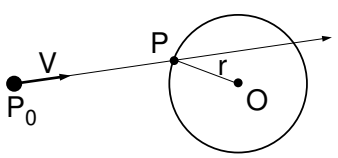
Ray−Sphere Intersection

- **Algebraic solution:**

Ray: $P = P_0 + tV$
Sphere: $|P - O|^2 - r^2 = 0$

substitute: $|\mathbf{P_0 + tV} - O|^2 - r^2 = 0$

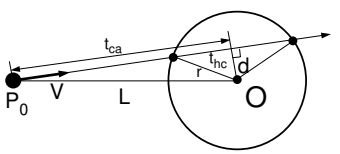
solve $At^2 + Bt + C = 0$
where $A = 1$
$B = 2\ V \bullet (P_0 - O)$
$C = |P_0 - O|^2 - r^2$



Ray−Sphere Intersection

- **Geometric solution:**

$L = O - P_0$
$t_{ca} = L \bullet V$
if ($t_{ca} < 0$) then no intersection
$d = L\bullet L - t_{ca}^2$
if ($d > r$) then no intersection
$t_{hc} = sqrt(r^2 - d^2)$
$t = t_{ca} - t_{hc}$ AND $t_{ca} + t_{hc}$

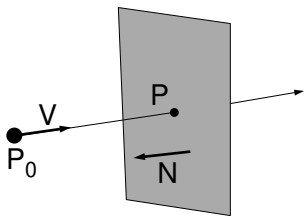


Ray–Plane Intersection

Ray: $P = P_0 + tV$
Plane: $P \bullet N + D = 0$

substitute: $(\mathbf{P_0 + tV}) \bullet N + D = 0$

solution: $t = -(P_0 \bullet N + D) / (V \bullet N)$



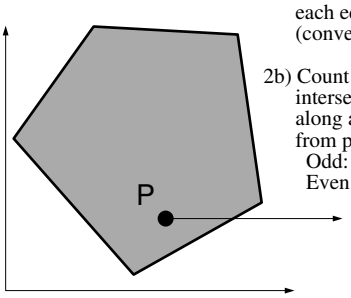
Ray–Polygon Intersection

1) P = ray–plane intersection

2) Check if polygon contains P

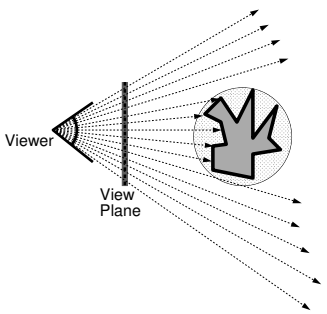
2a) Check if P is on left–side of each edge (convex only)

2b) Count edge intersections along any ray from point:
Odd: inside
Even: outside



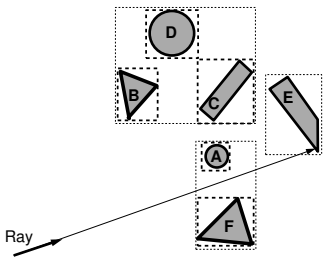
Ray Tracing Acceleration

- **Bounding volumes**
  - Check simple bounding volume for ray–surface intersections before checking complex shapes



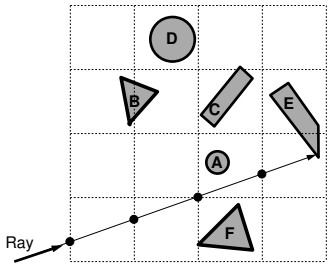
Ray Tracing Acceleration

- **Bounding volume hierarchies**
  - Construct and check hierarchical bounding volumes



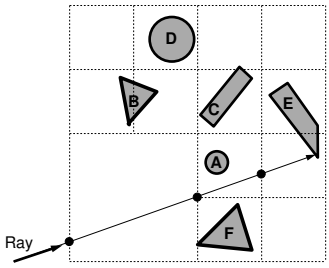
Ray Tracing Acceleration

- **Spatial data structures**
  - Uniform Grids:



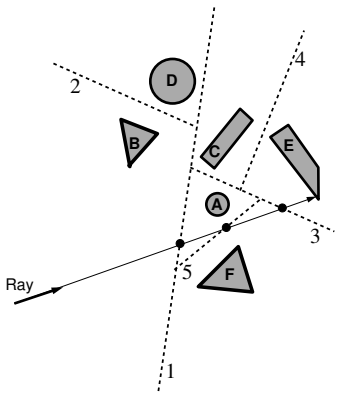
Ray Tracing Acceleration

- **Spatial data structures**
  - Octrees:



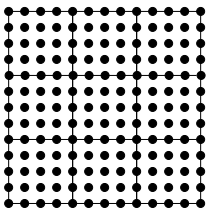
Ray Tracing Acceleration

- **Spatial data structures**
  - BSP trees:

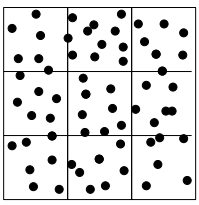


Antialiasing

- **Supersampling**
  - Multiple, evenly spaced samples per pixel



- **Adaptive sampling**
  - Randomized, directed samples



Summary

- **Ray casting**
- **Recursive ray tracing**
- **Ray trees**
- **Ray–surface intersections**
- **Spatial Subdivisions**
- **Antialiasing**