Computer Graphics

by Ruen-Rone Lee
ICL/ITRI

Wrap up from last week

- **Anti-alias Techniques**
  - **Super sampling**
  - **Multi sampling**
  - **Post processing**
  - **Temporal AA**

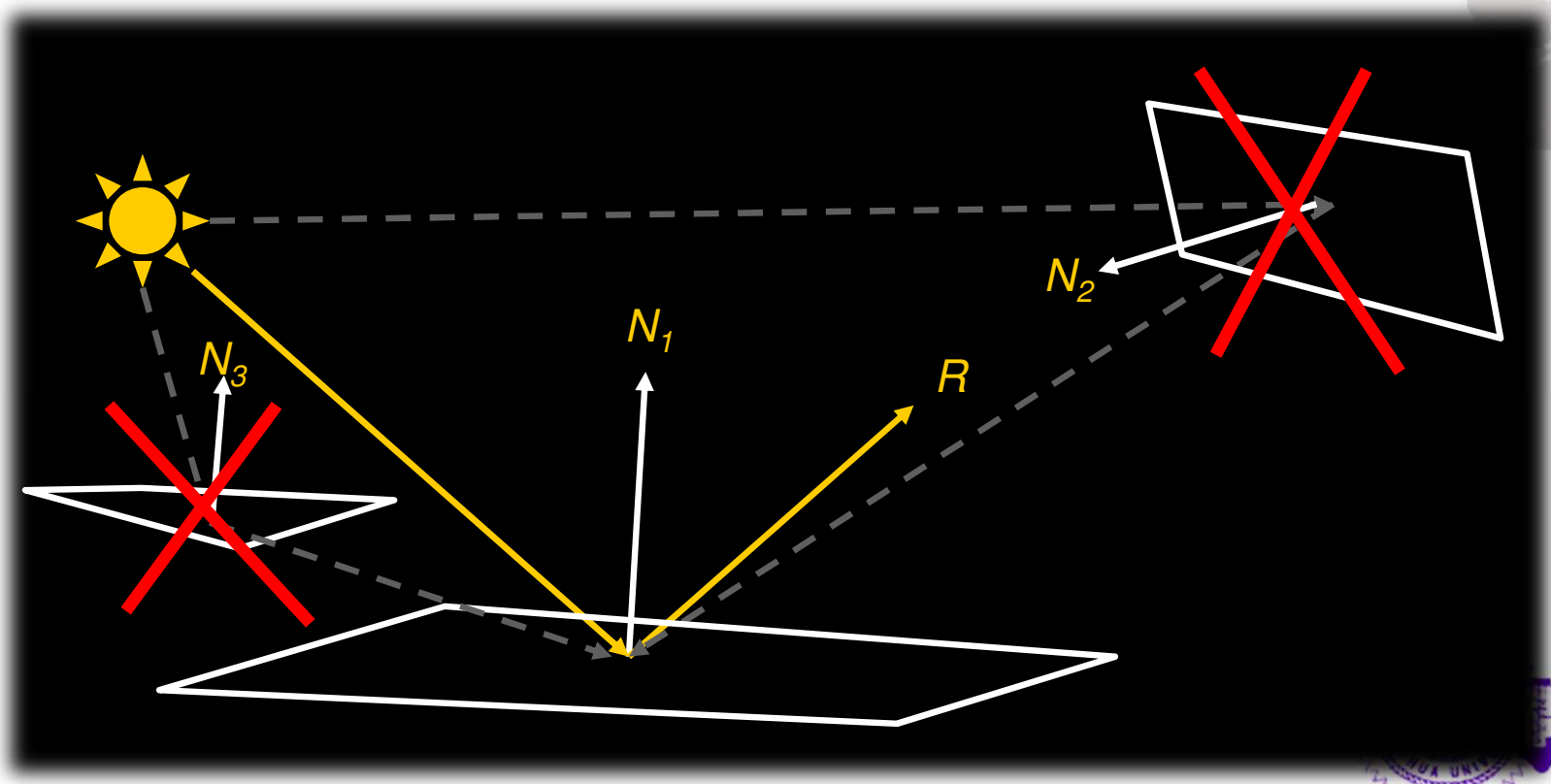
Global Illumination

Radiosity

Ray Tracing

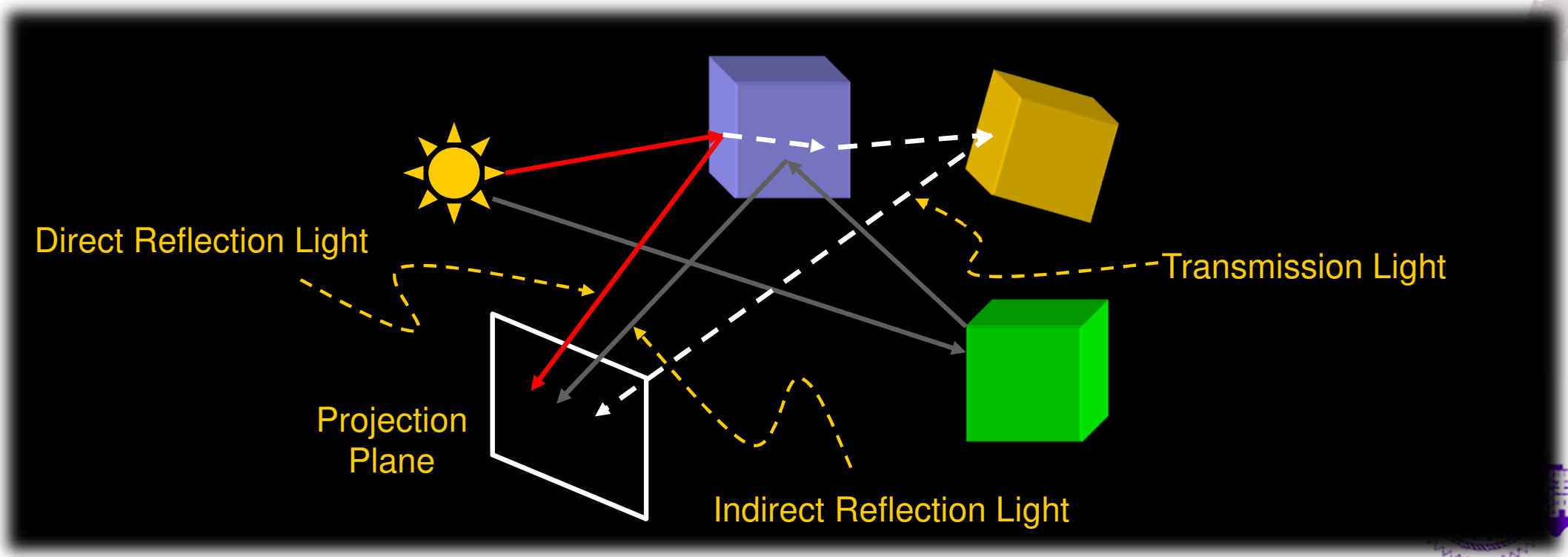
Local Illumination

- Local illumination counts only light sources directly on the shaded points
- No inter-reflection, no refraction, no realistic shadow



Global Illumination

- Global illumination counts not only the direct light sources, but also the indirect lights. Such as reflection or transmission lights from other objects

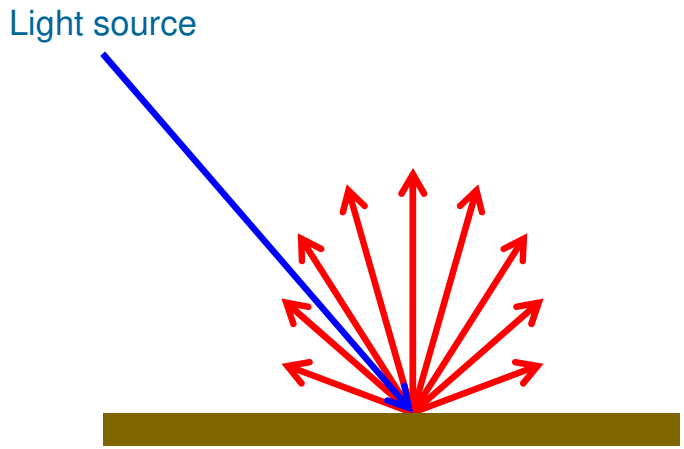


Global Illumination

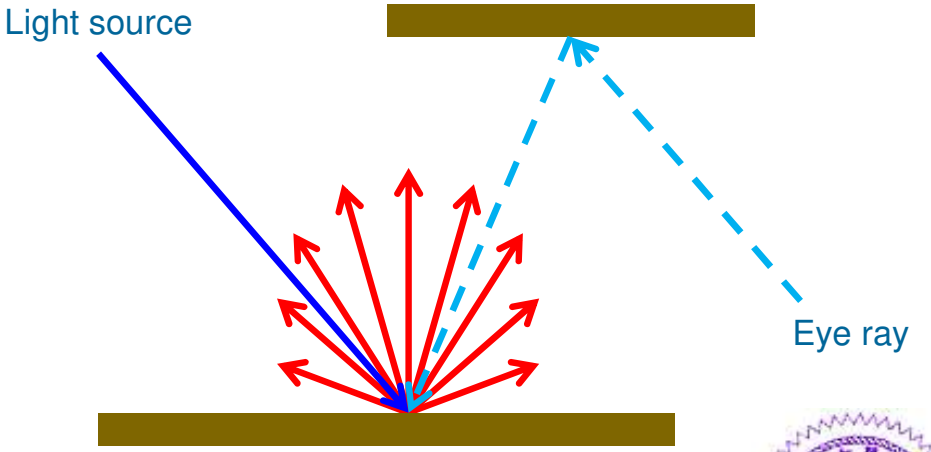
- Theoretically, reflections, refractions, and shadows are all examples of global illumination.
  - One object affects the rendering of another object (as opposed to an object being affected only by a direct light).
- In practice, however, only the simulation of diffuse inter-reflection or caustics is called global illumination.

Global Illumination

- **Diffuse inter-reflection**
  - **A process whereby light reflected from an object strikes other objects in the surrounding area and illuminating them**
  - **If the diffuse surface is colored, the reflected light is also colored**



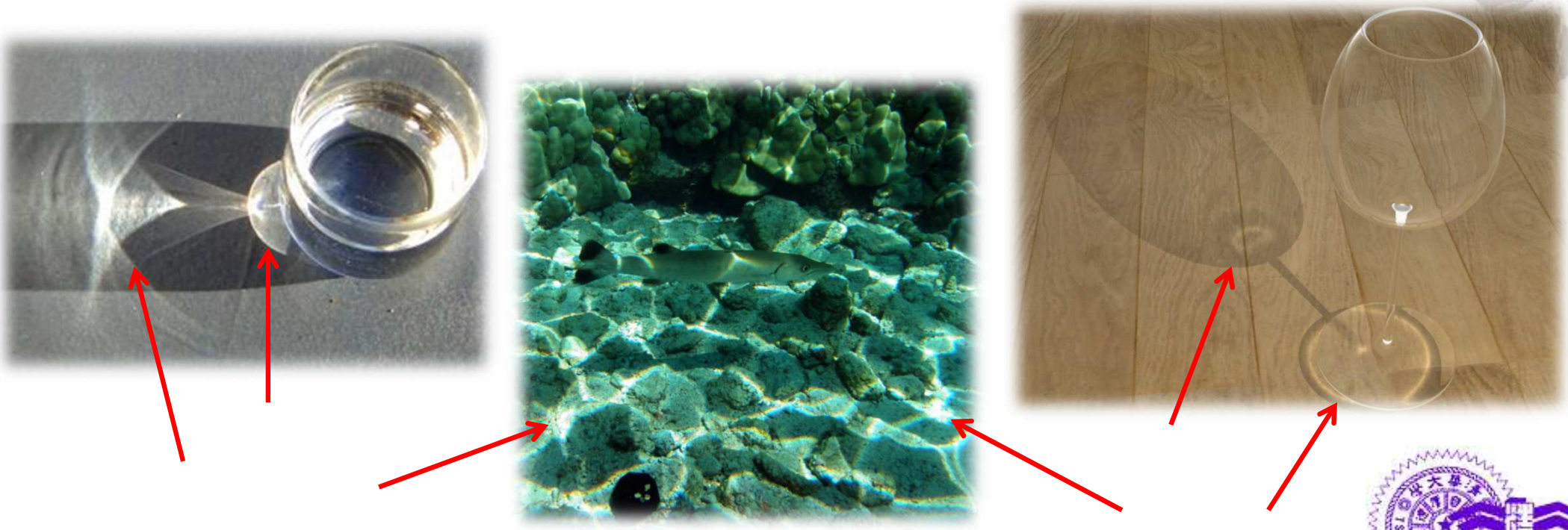
Diffuse Reflection



Diffuse Inter-reflection

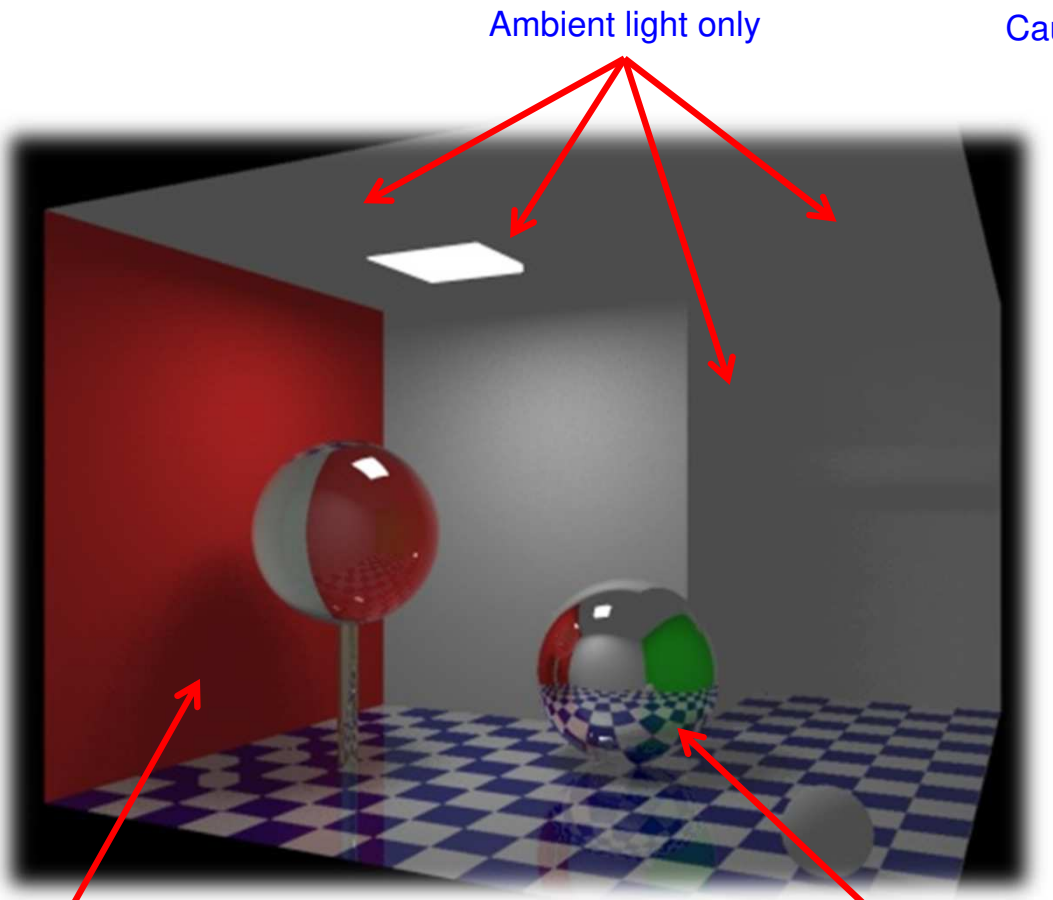
Global Illumination

- **Caustic**
  - **A caustic is the envelope of light rays reflected or refracted by a curved surface or object, or the projection of that envelope of rays on another surface.**

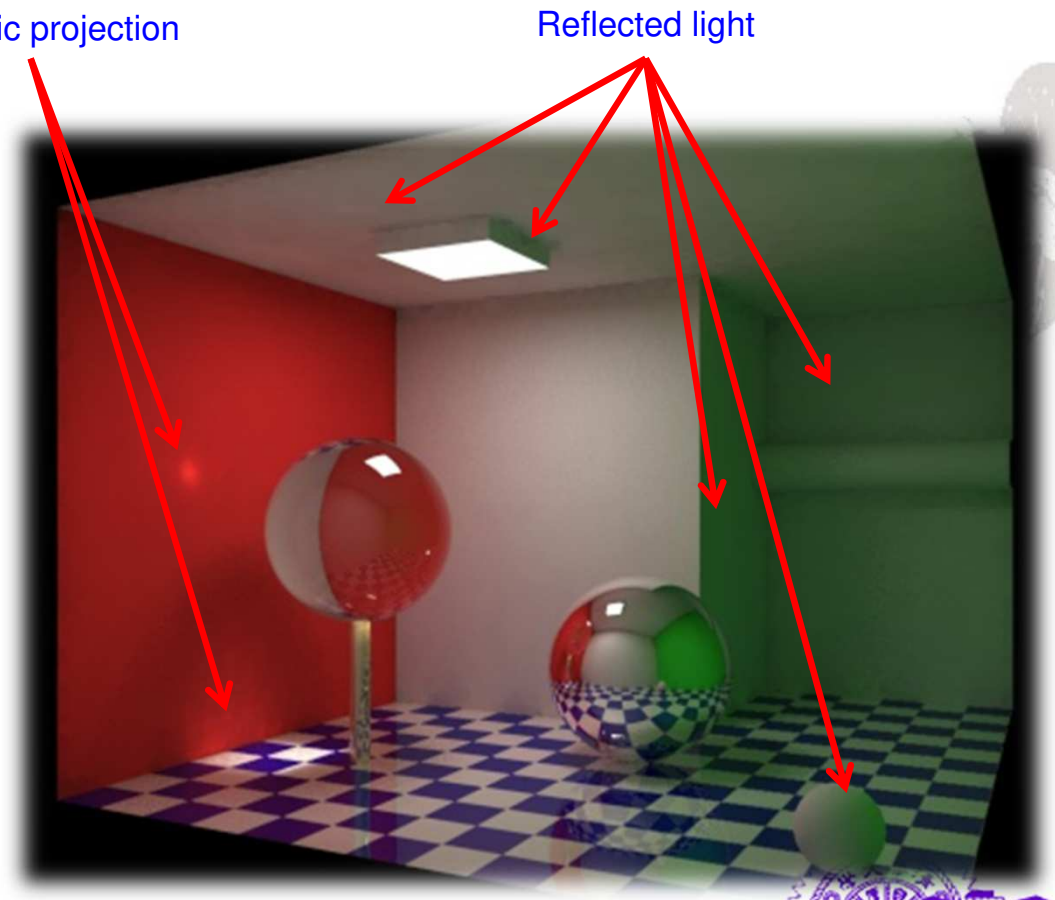


Global vs. Local Illumination

- **Comparison**



Local Illumination



Global Illumination

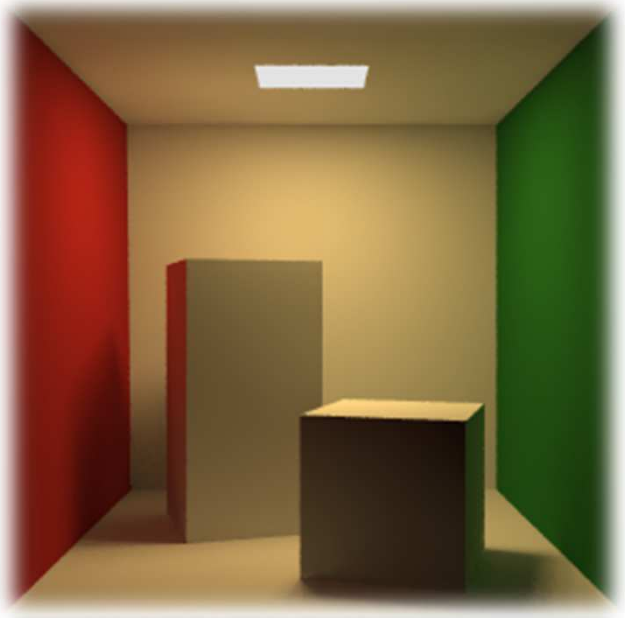
Examples of Global Illumination

- Radiosity
- Ray tracing
- Beam tracing
- Cone tracing
- Path tracing
- Metropolis light transport
- Ambient occlusion
- Photon mapping
- Image-based lighting

Radiosity

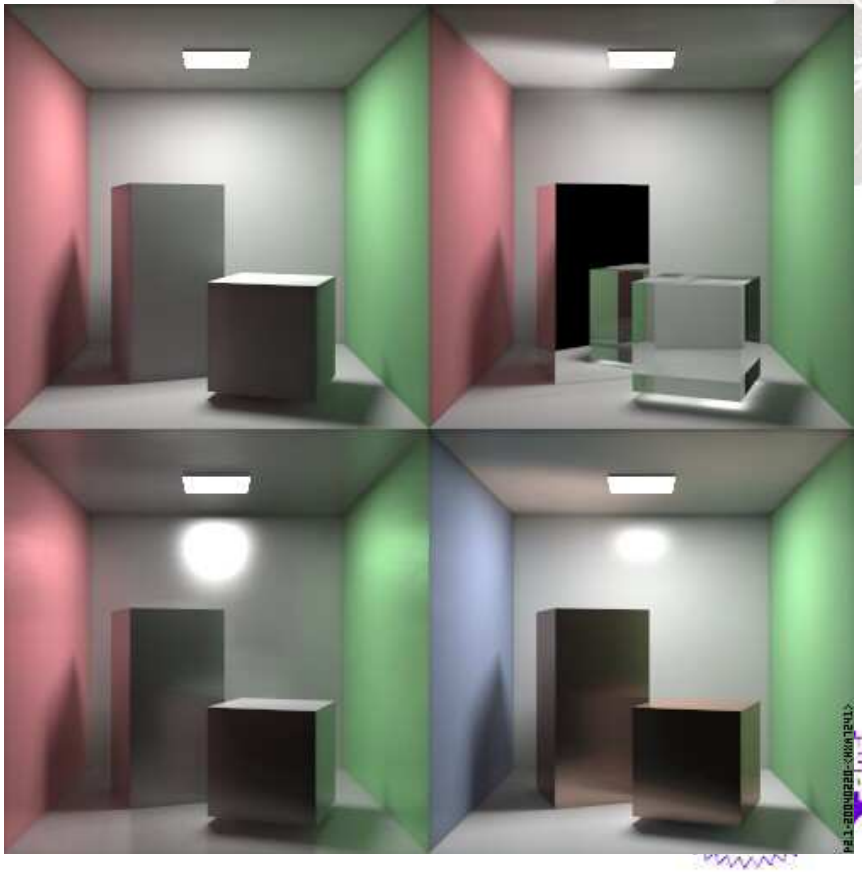
Radiosity

- The illumination arriving at the eye comes not just from the light sources, but all the scene surfaces interacting with each other as well.

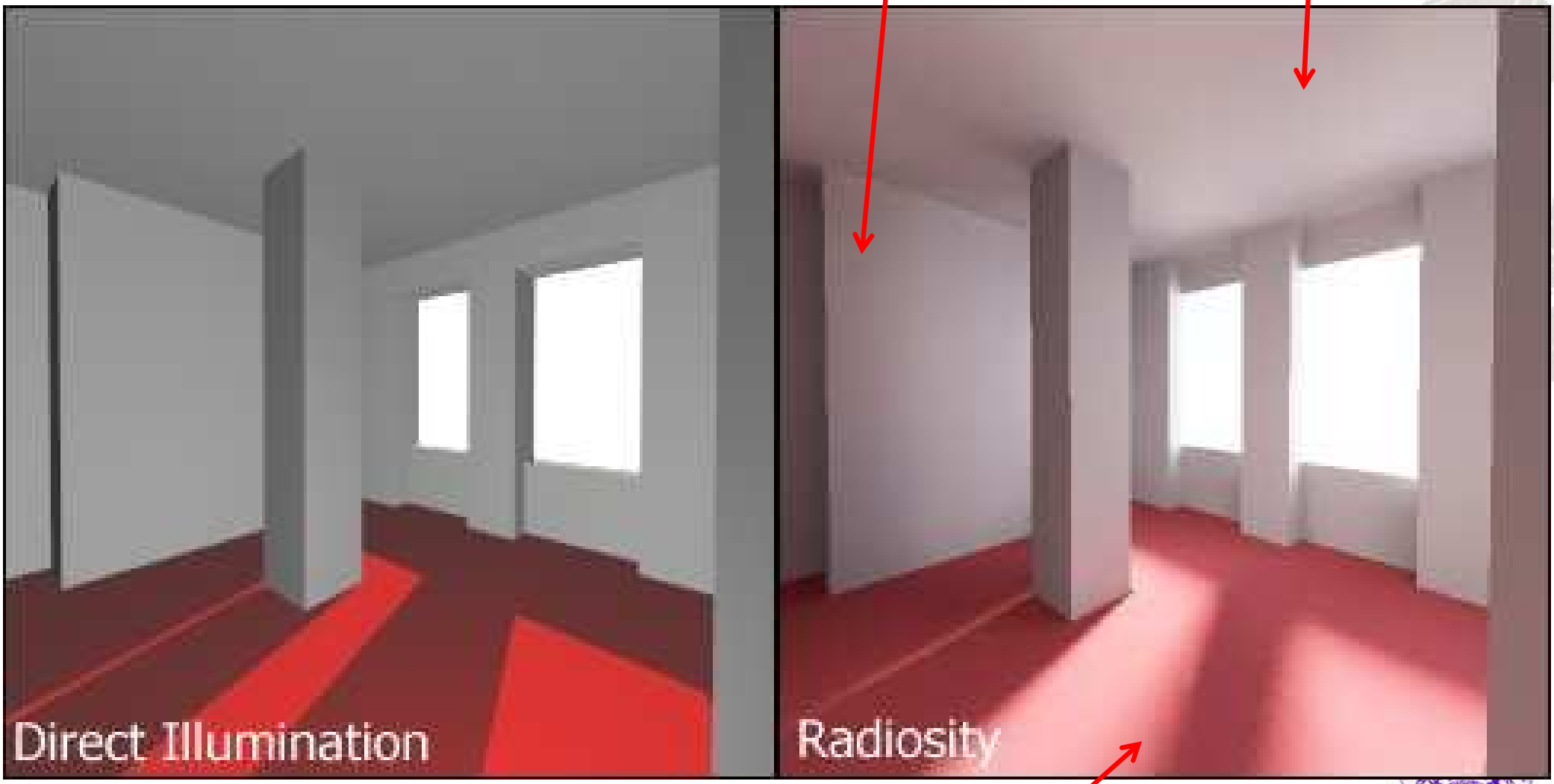


The Cornell Box

Variation



Radiosity


Red carpet reflection
Subtle lighting effects
Direct Illumination
Radiosity
Soft shadow

Radiosity

- The radiosity of a surface is the rate at which energy leaves that surface
- It includes the energy emitted by a surface as well as the energy reflected from other surfaces
- Radiosity methods allow the intensity of radiant energy arriving at a surface to be computed. These intensities can then be used to detemine the shading of the surface.

Form Factor (View Factor)

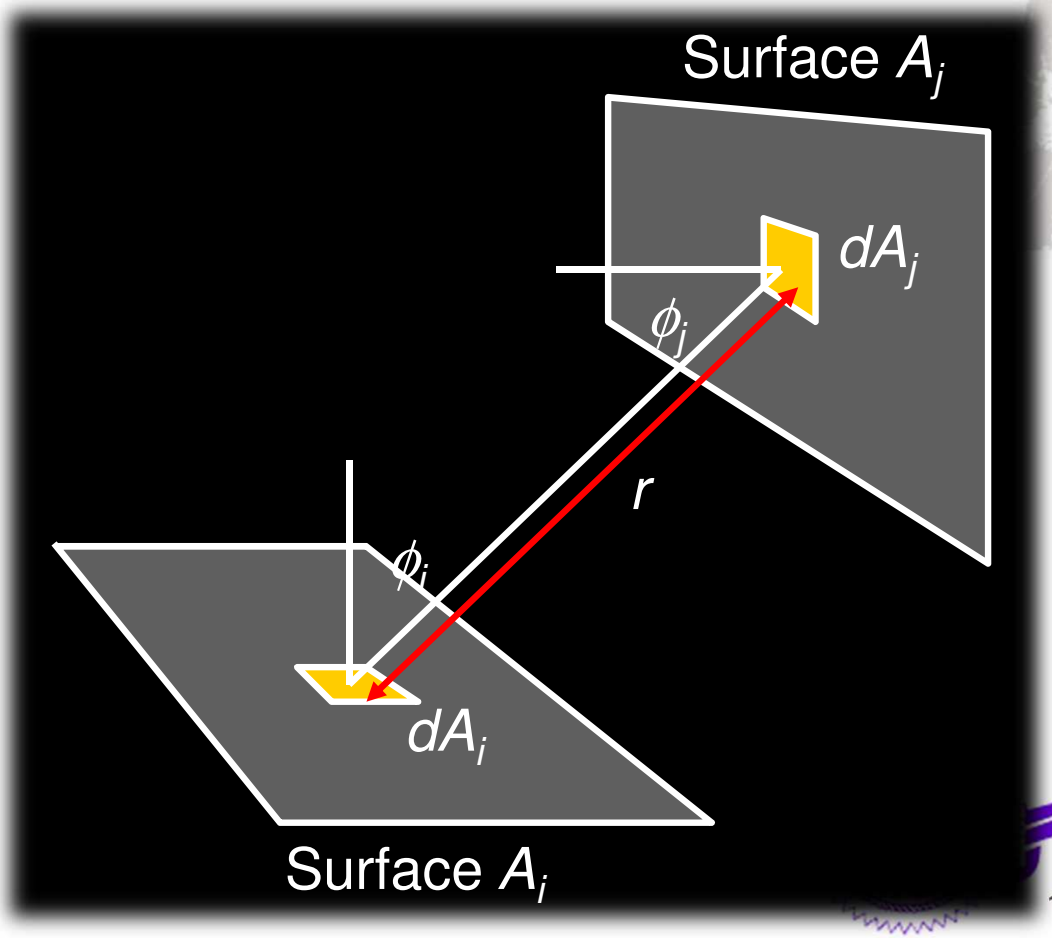
- The form factor is defined as the fraction of energy leaving one surface that reaches another surface

Form factor: $dA_j \rightarrow dA_i$

$$F_{ij} = \frac{\cos\phi_i \cos\phi_j}{\pi |r|^2} dA_j$$

Form factor: $A_j \rightarrow A_i$

$$F_{ij} = \frac{1}{A_i} \int_{A_i} \int_{A_j} \frac{\cos\phi_i \cos\phi_j}{\pi |r|^2} dA_j dA_i$$

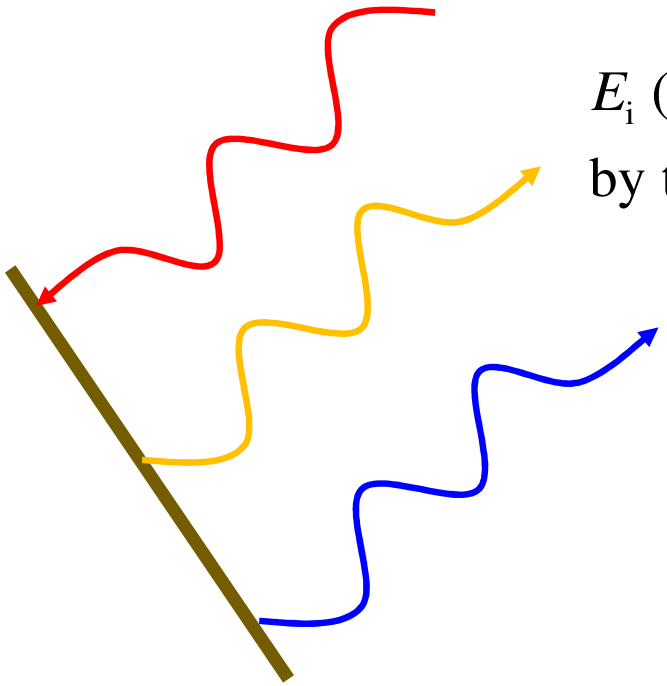


The Radiosity Equation

$\sum B_j F_{ij}$ (energy reaching this surface from other surfaces)

E_i (energy emitted by this surface)

$\rho_i \sum B_j F_{ij}$ (energy reflected by this surface)



$$B_i = E_i + \rho_i \sum B_j F_{ij}$$

B_i = Radiosity of surface i

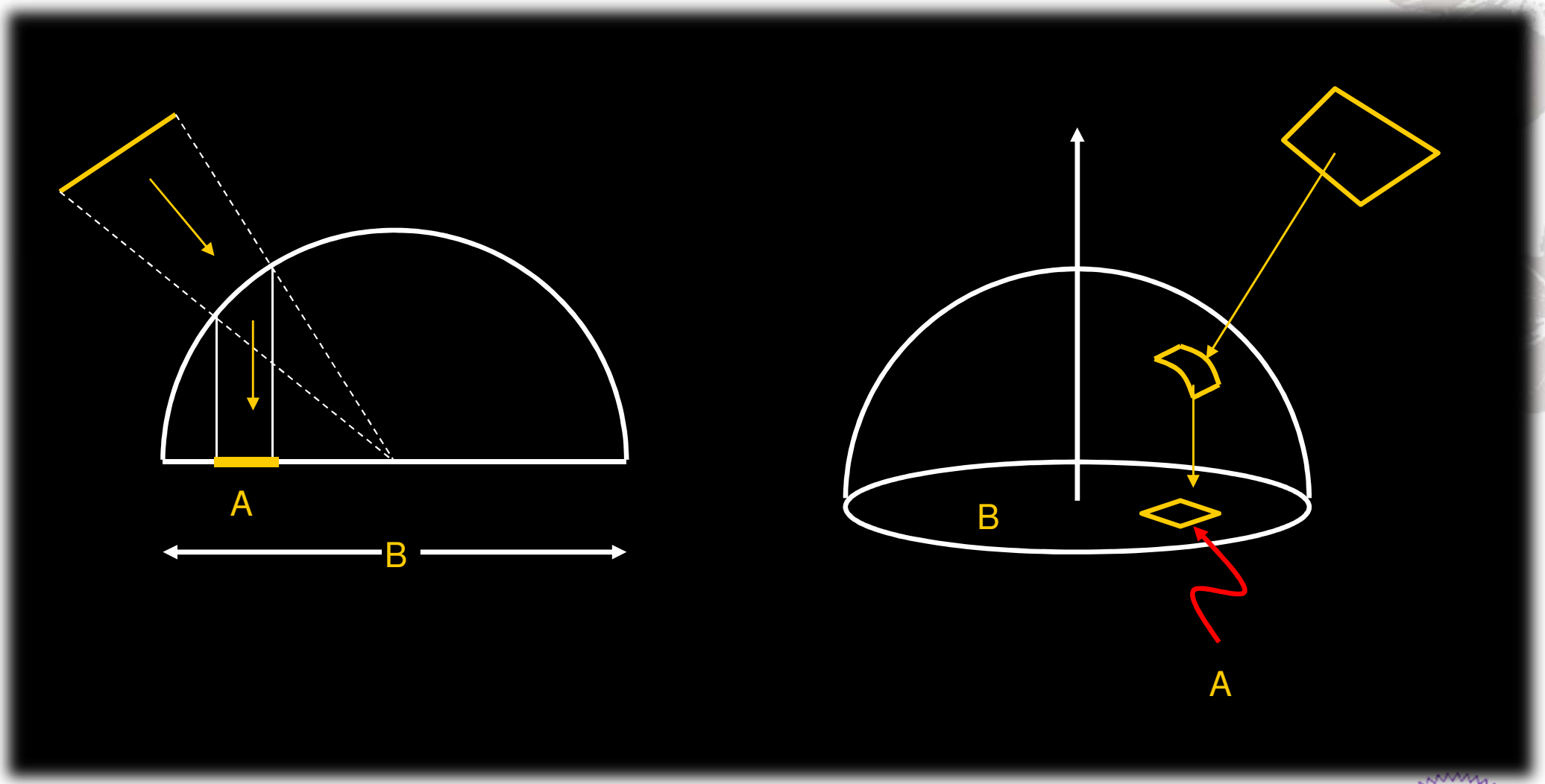
E_i = Emissivity of surface i

ρ_i = Reflectivity of surface i

B_j = Radiosity of surface j

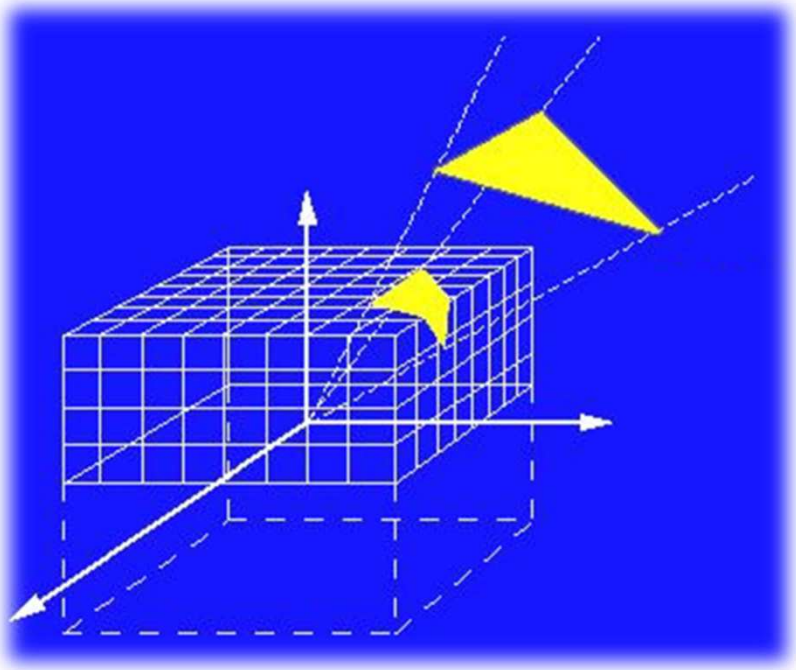
F_{ij} = Form Factor for the radiation leaving surface j and hitting surface i

Hemisphere

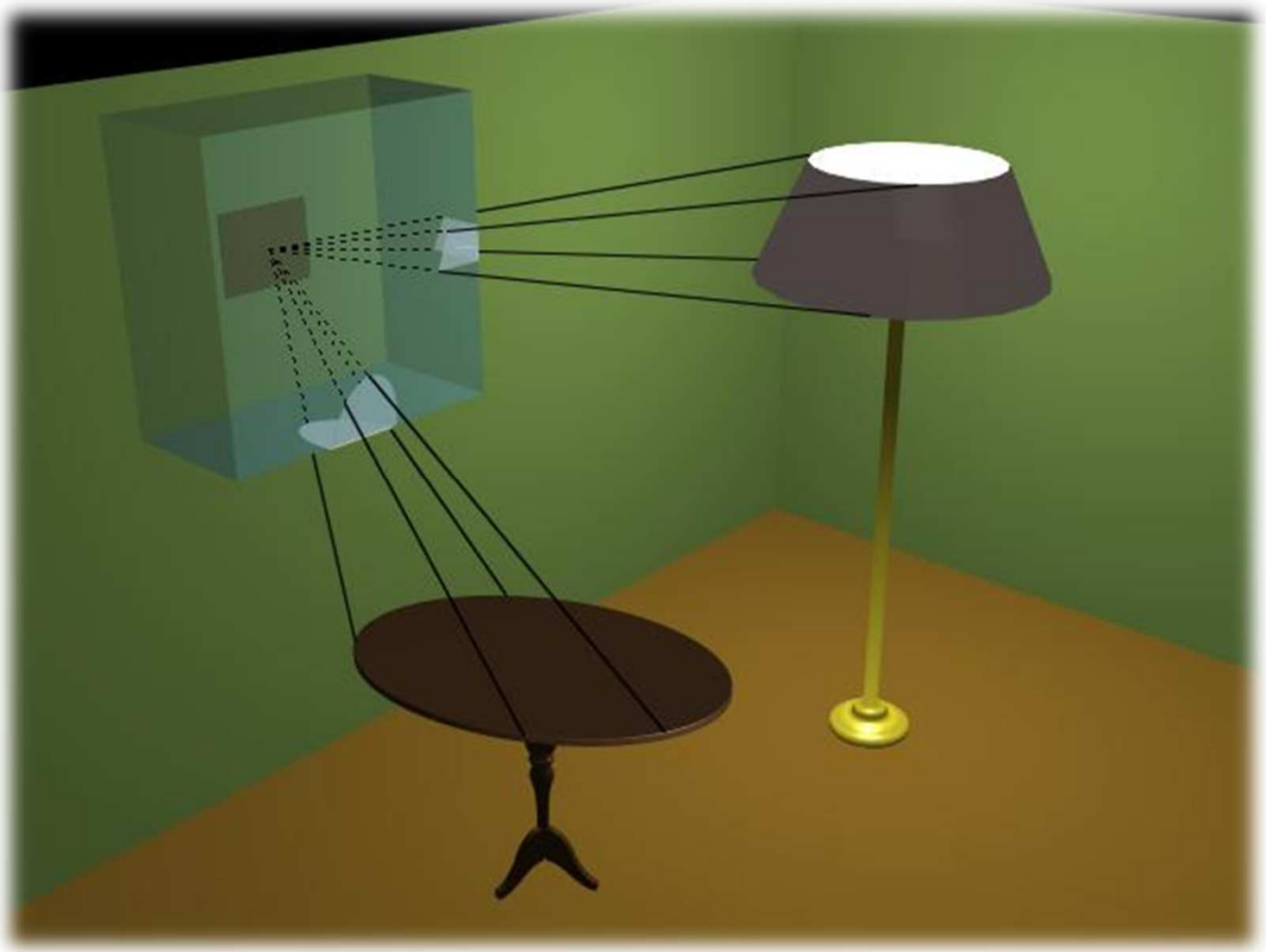


Hemicube Approximation

- Project each patches onto the hemicube
- Each pixel has pre-computed form factor
- The form factor is the sum of the pixels it overlaps



Hemicube Approximation



Resolving Radiosity Equation

- **Solving Full Matrix**

$$\begin{bmatrix} 1-\rho_1 F_{11} & -\rho_1 F_{12} & \cdots & -\rho_1 F_{1n} \\ -\rho_2 F_{21} & 1-\rho_2 F_{22} & \cdots & -\rho_2 F_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ -\rho_n F_{n1} & -\rho_n F_{n2} & \cdots & 1-\rho_n F_{nn} \end{bmatrix} \begin{bmatrix} B_1 \\ B_2 \\ \vdots \\ B_n \end{bmatrix} = \begin{bmatrix} E_1 \\ E_2 \\ \vdots \\ E_n \end{bmatrix}$$

ρ_i is the reflectivity of surface i

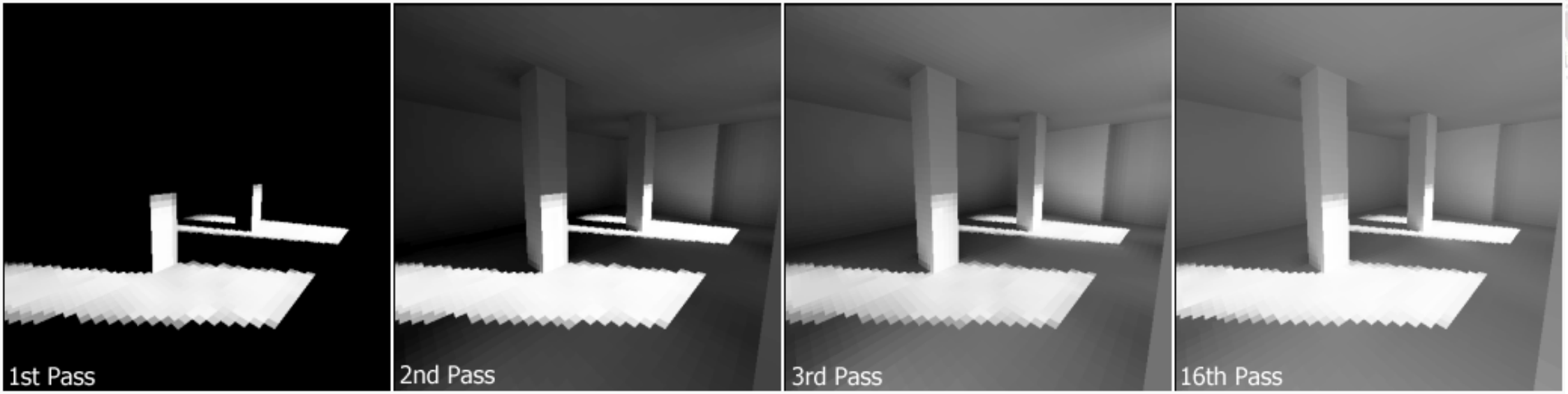
F_{ij} is the form factor from surface j to surface i

B_i is the radiosity of surface i

E_i is the emission of surface i

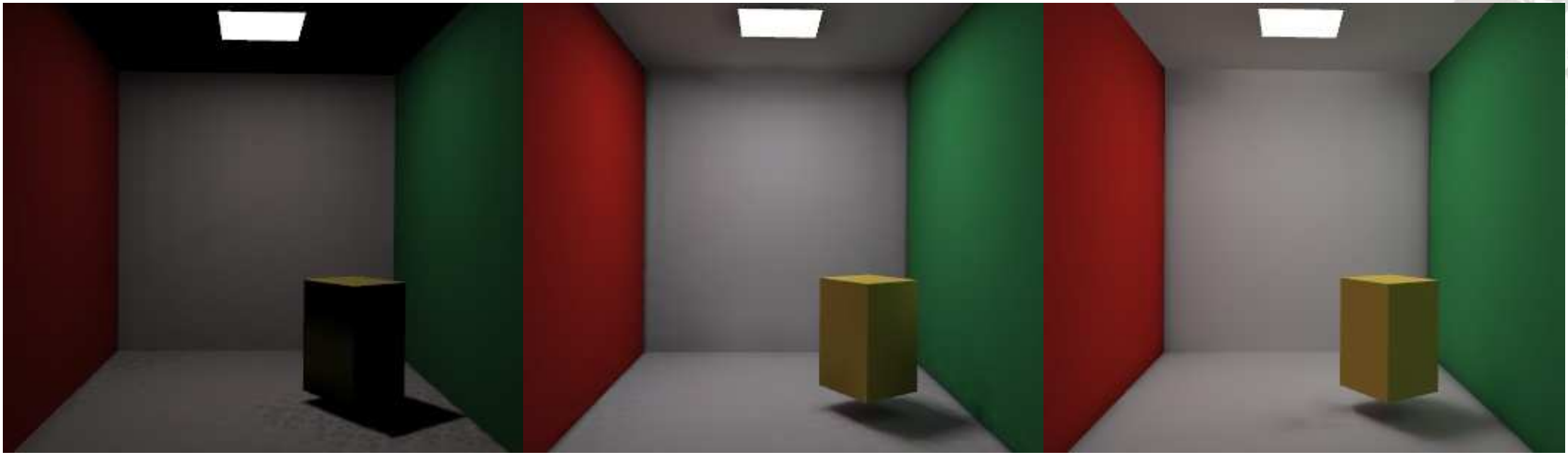
Iterations of Radiosity Algorithm

- Iteratively solves the radiosity equation by "shooting" light from the patch with the most error at each step



Iterations of Radiosity Algorithm

- Iteratively solves the radiosity equation by "shooting" light from the patch with the most error at each step



1 iteration

32 iterations

320 iterations

Progressive Solution

PROGRESSIVE SOLUTION

The above images show increasing levels of global diffuse illumination. From left to right: 0 bounces, 1 bounce, 3 bounces.

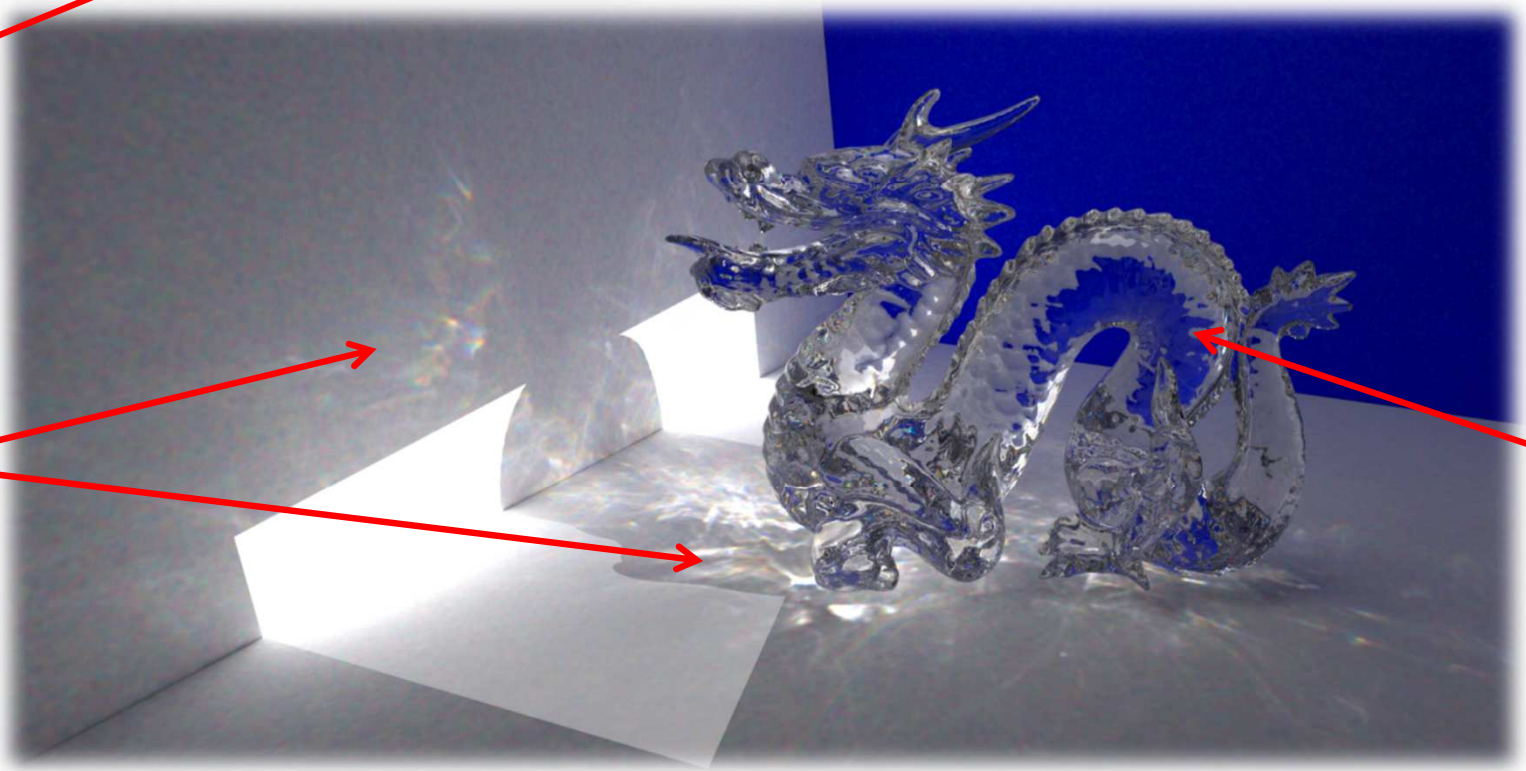
Characteristic of Radiosity

- An object space algorithm
- View independent (can be pre-computed)
- Render perfect diffuse scenes
- Area light sources (polygonal patches)
- Ideas came from the conservation of energy

Ray Tracing

Reflection

Scattering

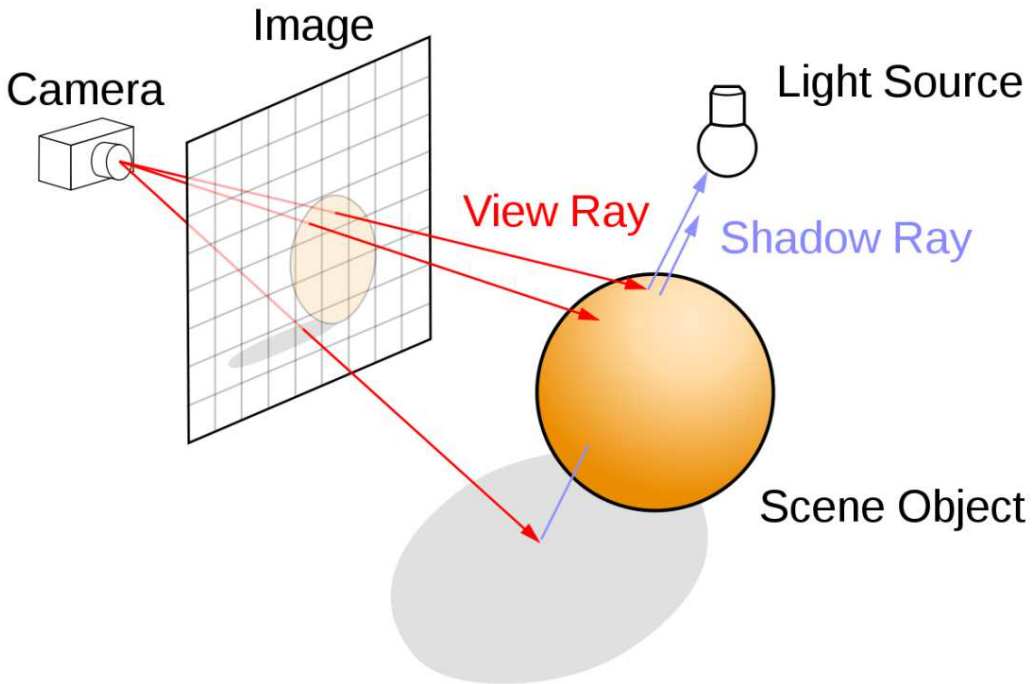


Dispersion

Refraction

Ray Tracing

- A technique for generating an image by tracing the path of light through pixels in an image plane and simulating the effects of its encounters with virtual objects.

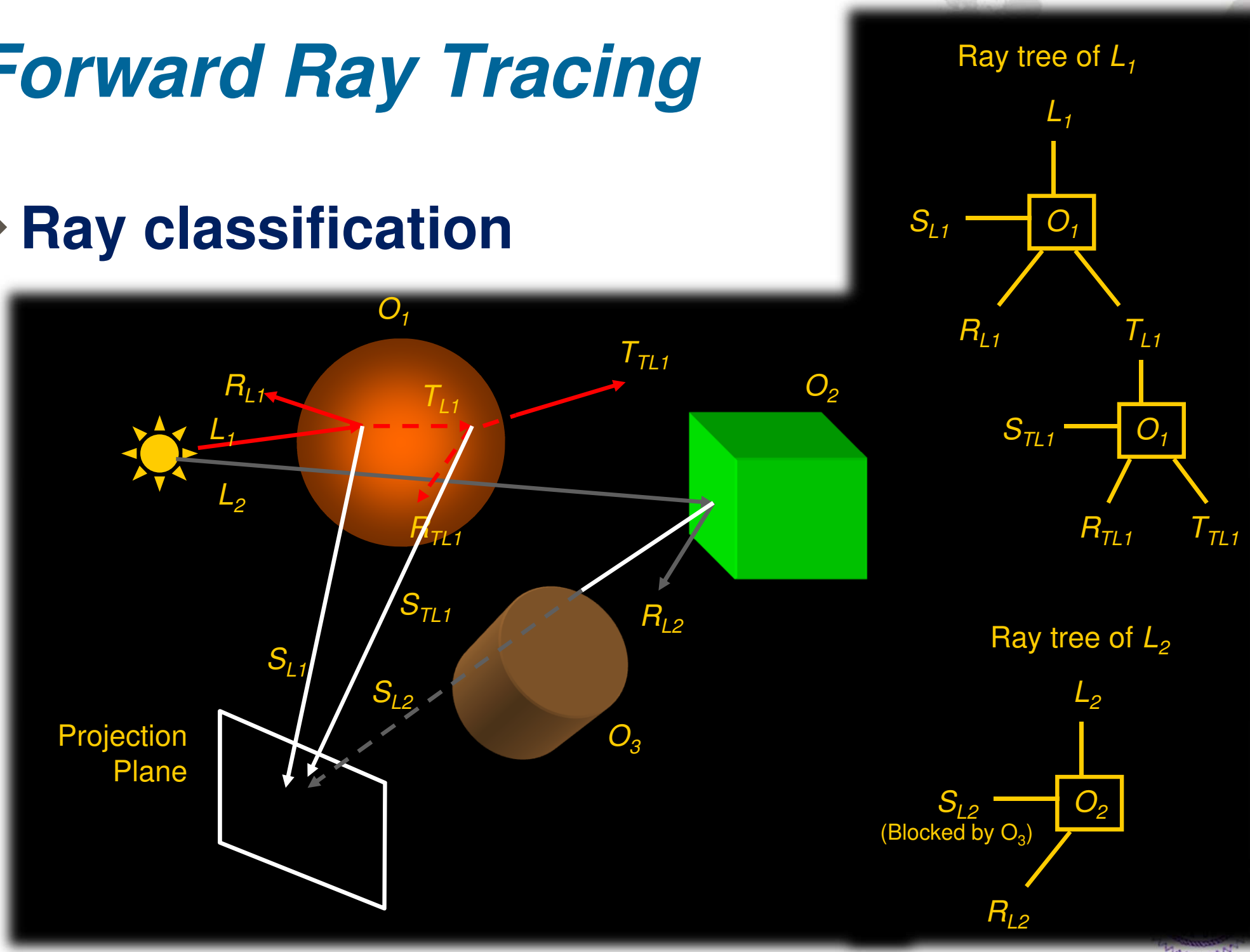


Forward Ray Tracing

- Light rays (primary rays) emitted from light sources
- Secondary rays including a shadow ray, a reflection ray, and a refraction ray (if translucent object) are generated for each light-object intersection
- Shadow ray is used to check whether the intersection point is illuminated or in shadow
- Reflection and refraction rays are traced recursively as if they are new light rays

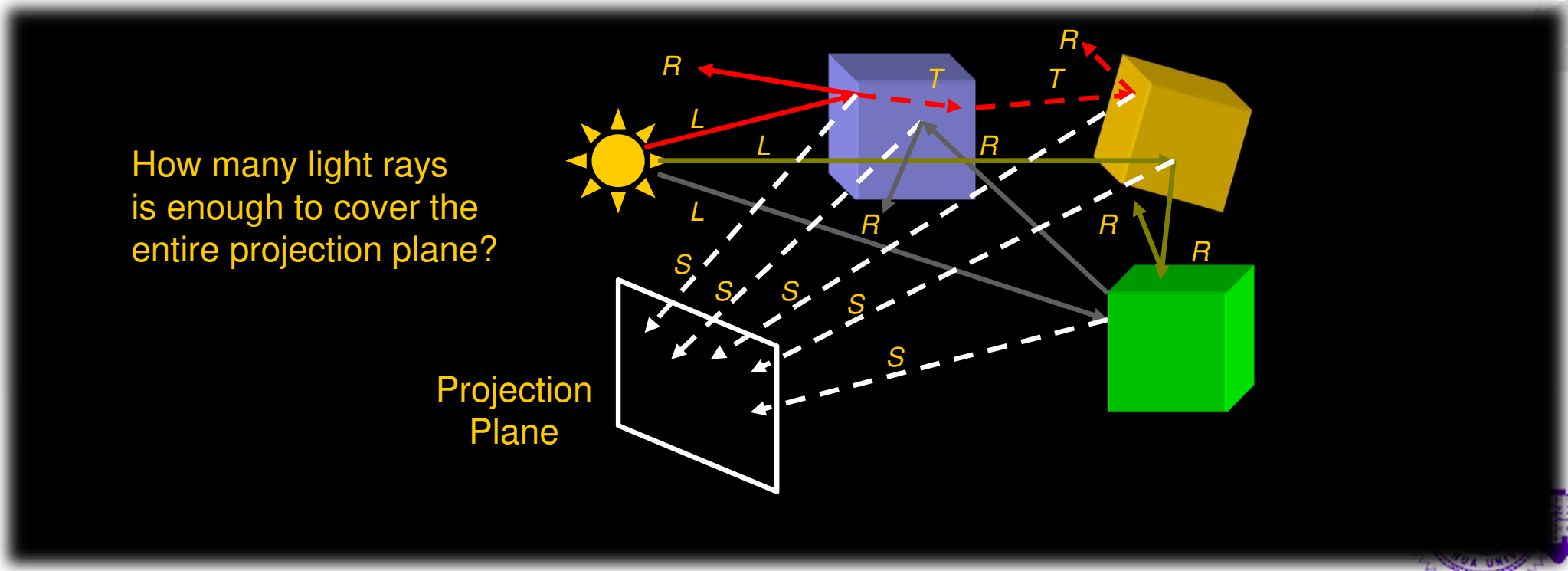
Forward Ray Tracing

- Ray classification



Forward Ray Tracing

- Trace light rays emitted from light sources
- Light rays bounced between objects in scene
- Accumulate the contribution of each light rays if they hit the projection plane

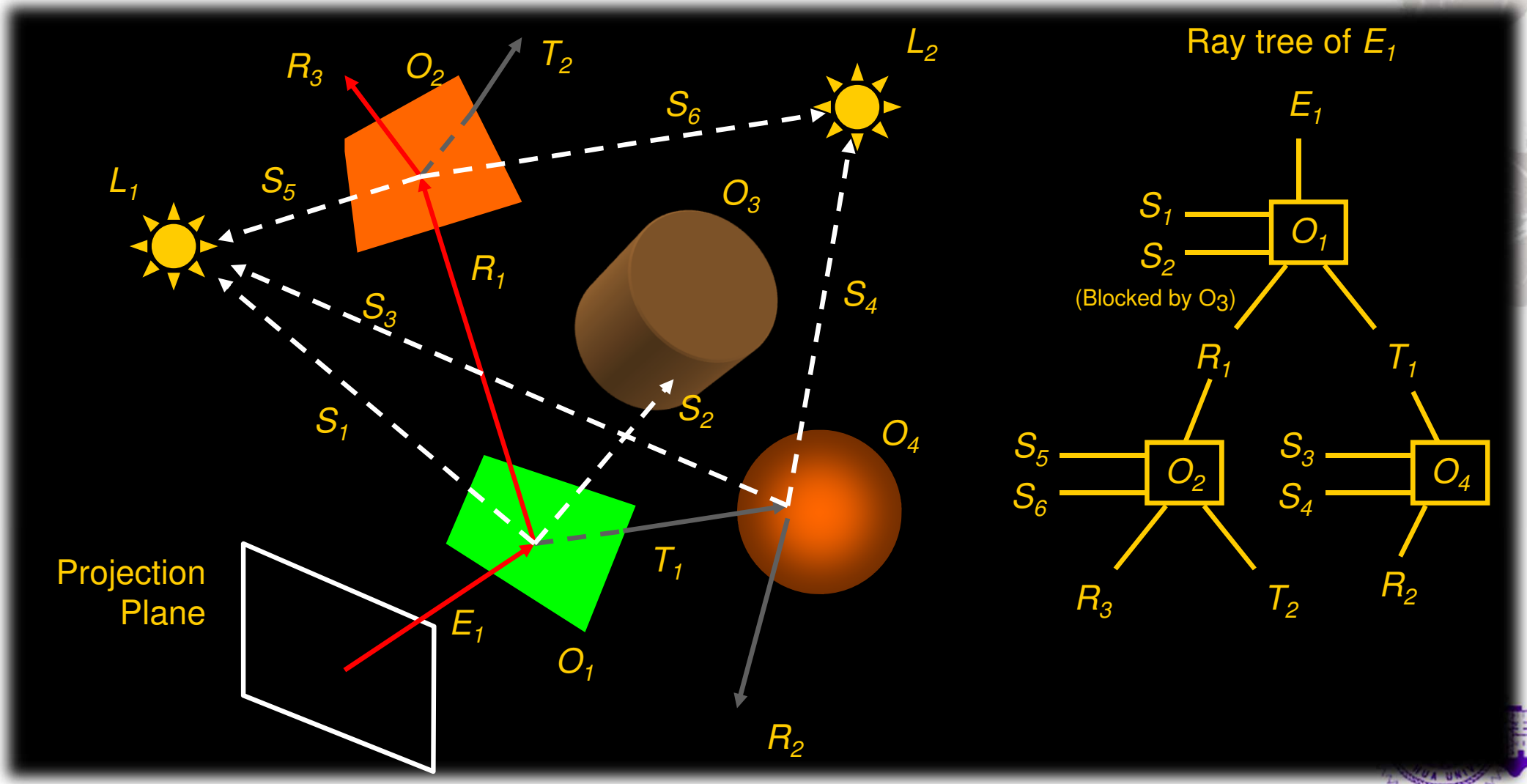


Backward Ray Tracing

- Eye rays (primary rays) emitted from each pixel of projection plane
- Secondary rays including a shadow ray, a reflection ray, and a refraction ray (if translucent object) are generated for each ray-object intersection
- Shadow ray is used to check whether the intersection point is illuminated or in shadow
- Reflection and refraction rays are traced recursively as if they are new eye rays

Backward Ray Tracing

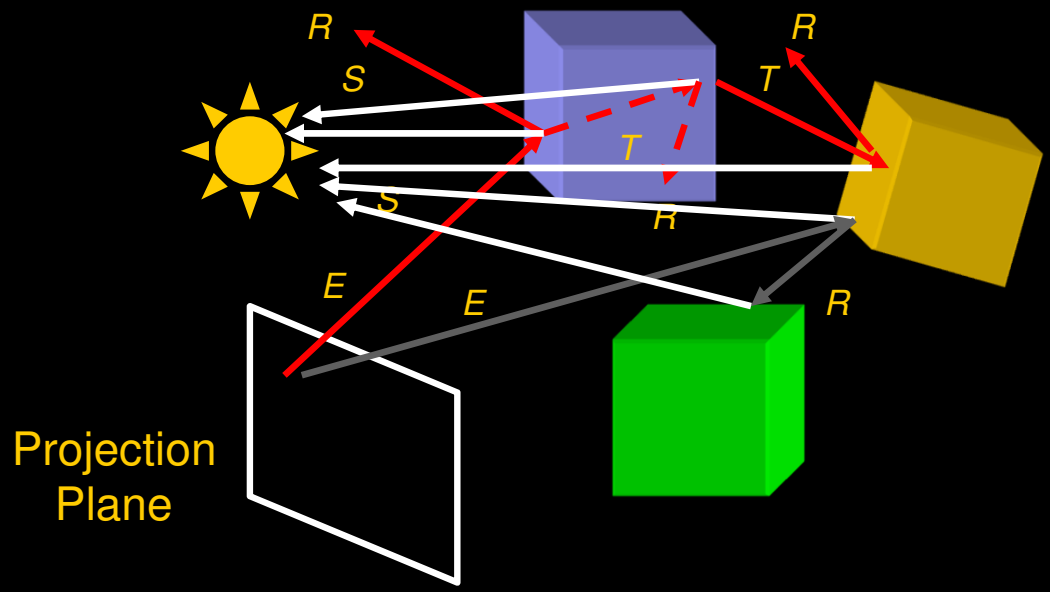
- **Ray classification**



Backward Ray Tracing

- Trace eye rays emitted from projection plane
- Eye rays bounced between objects in scene
- Accumulate the contributions of each shadow rays if they can reach the light sources

If the level of secondary rays is limited, then the algorithm can be terminated in finite time



Efficiency of Ray Tracing

- **Reduce the number of ray-object intersections**
  - **Bounding Box**
  - **Space Partition**
- **Parallel Processing**

Characteristic of Ray Tracing

- An image space algorithm
- View dependent
- Render scenes with perfect specular reflection and refraction
- Point light sources
- Ideas came from the path of light flow

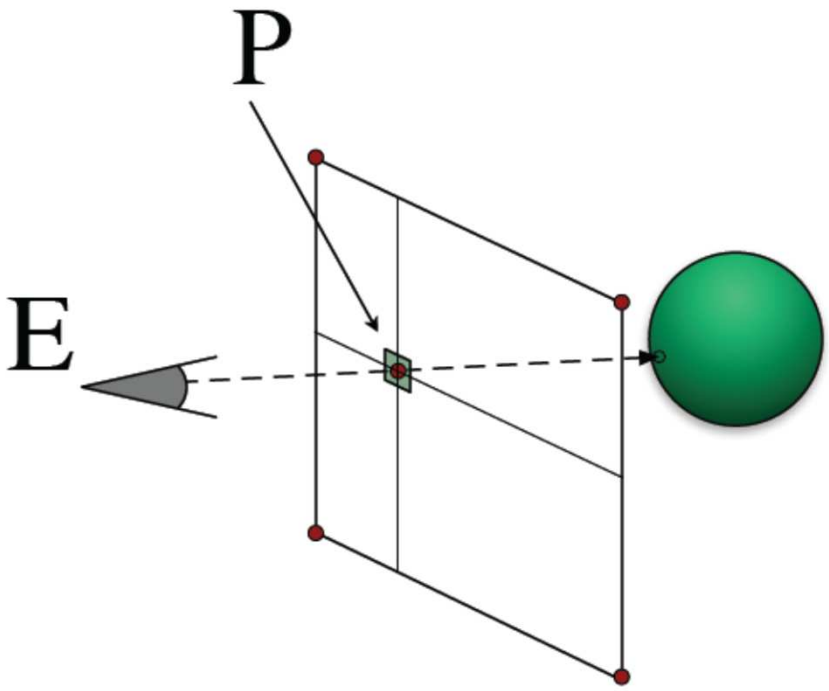
Building Eye Rays

- **Ray equation**

$$\mathbf{R}(t) = \mathbf{E} + t\,(\mathbf{P} - \mathbf{E})$$

$$t \in [1. . .+\infty]$$

  - **Through eye at $t = 0$**
  - **Through pixel center at $t = 1$**



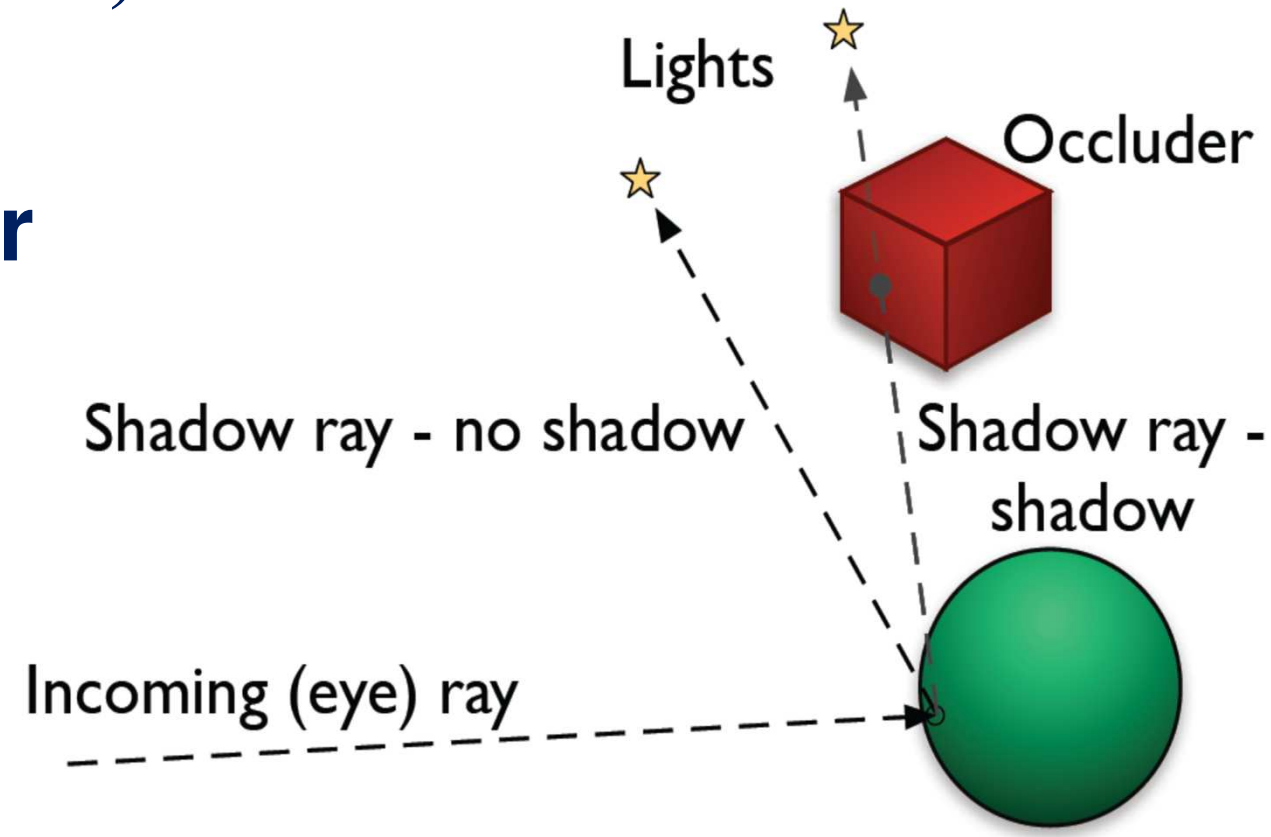
Shadow Rays

- Detect shadow by rays to light sources

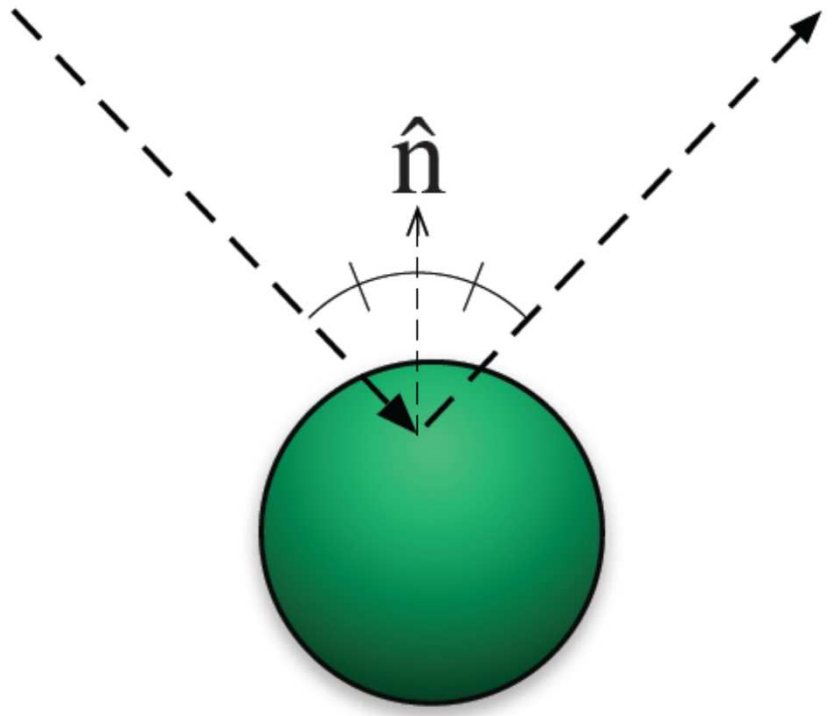
$$\mathbf{R}(t) = \mathbf{S} + t\,(\mathbf{L} - \mathbf{S})$$

$$t \in [\varepsilon \ldots 1)$$

- Test for occluder
  - Shade normally if no occluder



Reflection Rays



- **Recursive shading**

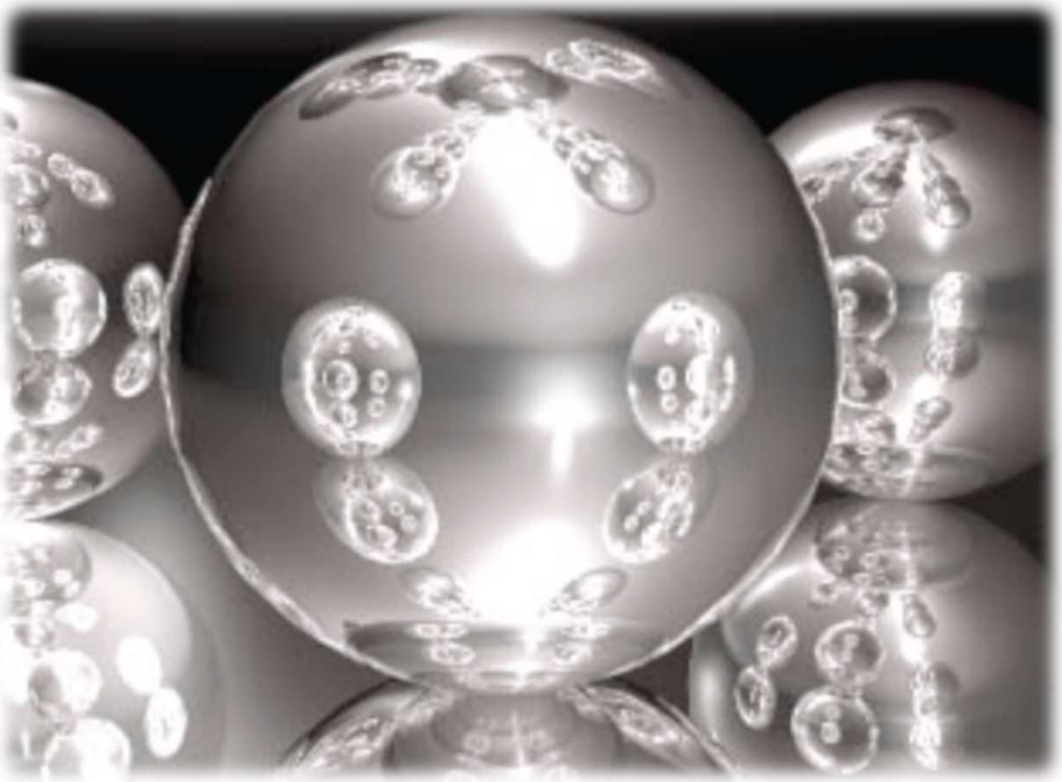
$$\mathbf{R}(t) = \mathbf{S} + t\,\mathbf{B}$$

$$t \in [\varepsilon \ldots +\infty)$$

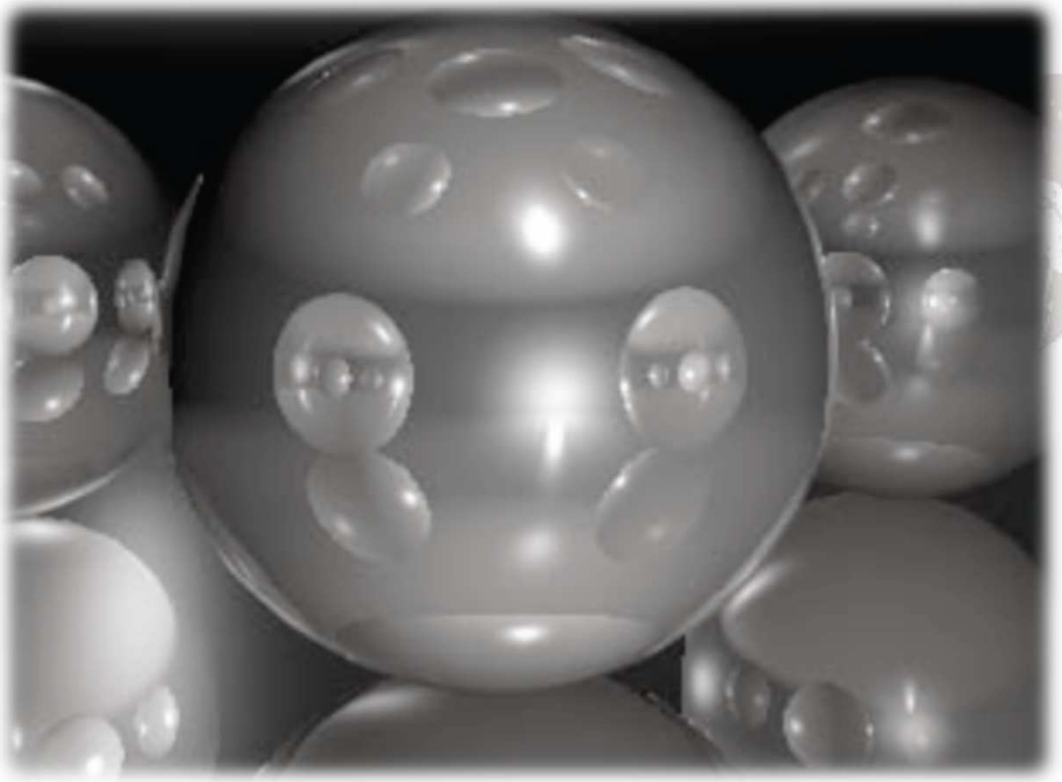
  - **Ray bounces off object**
  - **Treat bounce rays (mostly) like eye rays**
  - **Shade bounce ray and return color**
  - **Add color to shading at original point**

Reflection Rays

- Recursion Depth

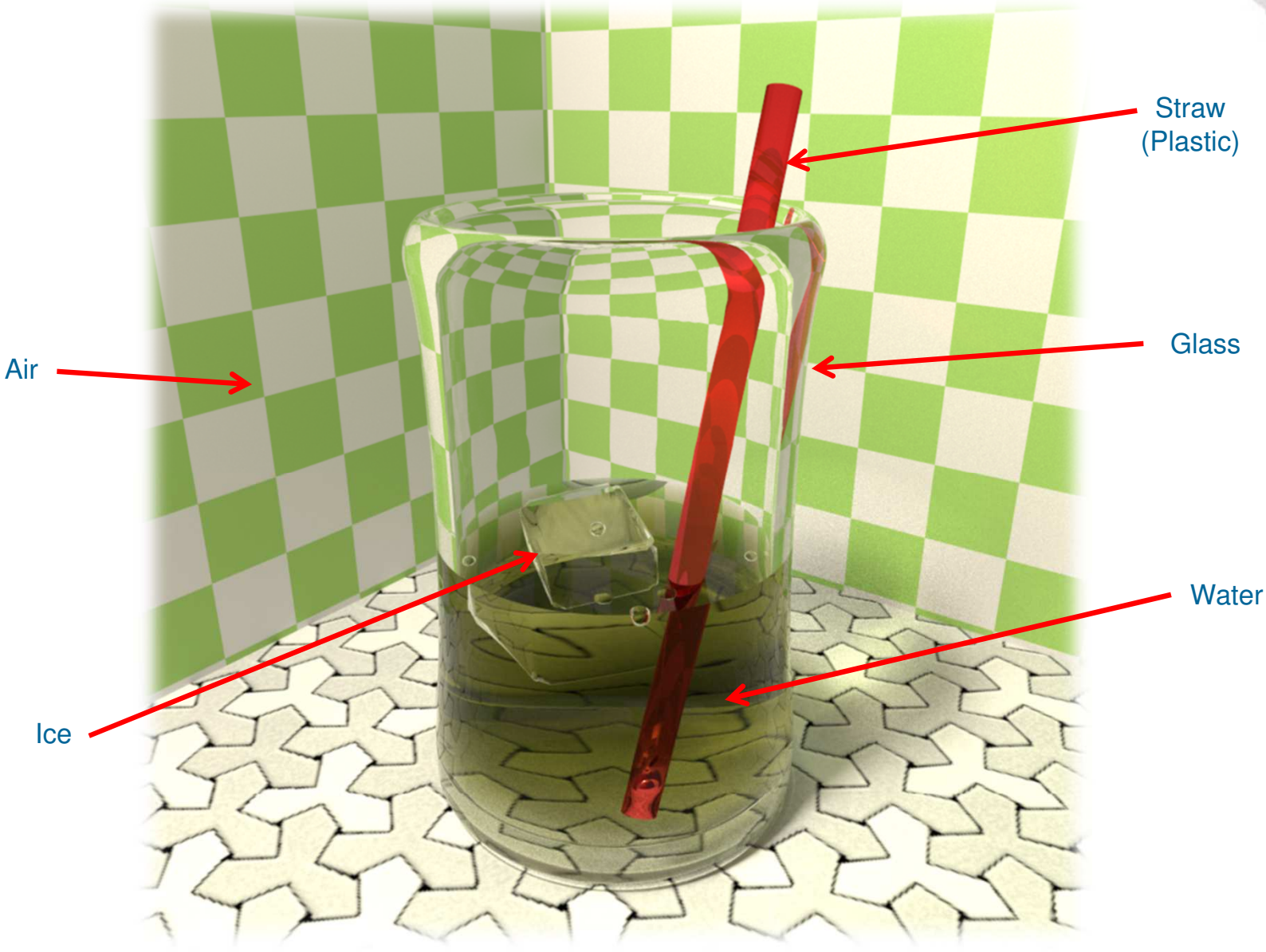


High recursion depth



Low recursion depth

Refraction Rays

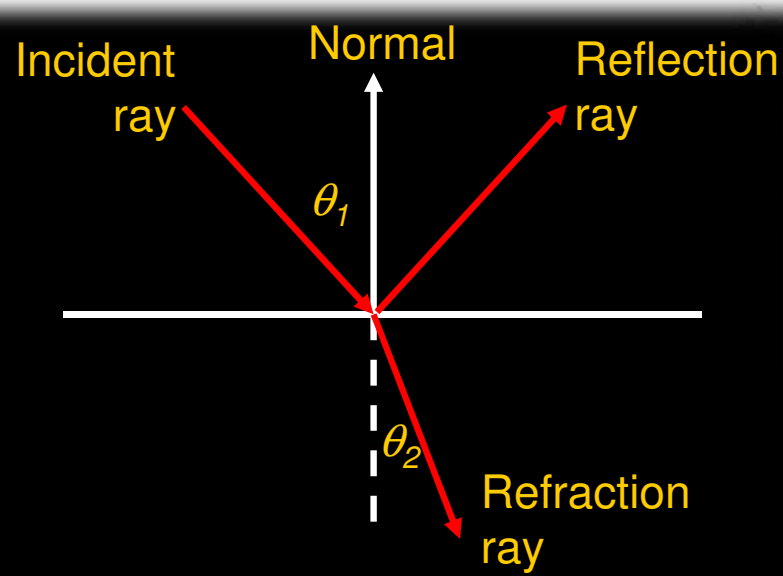


Refraction Rays

- **Transparent materials bend light**
  - **Snell's Law**

$$\frac{\sin\theta_1}{\sin\theta_2} = \eta_{21} = \frac{\eta_2}{\eta_1}$$

where η is the indices of refraction

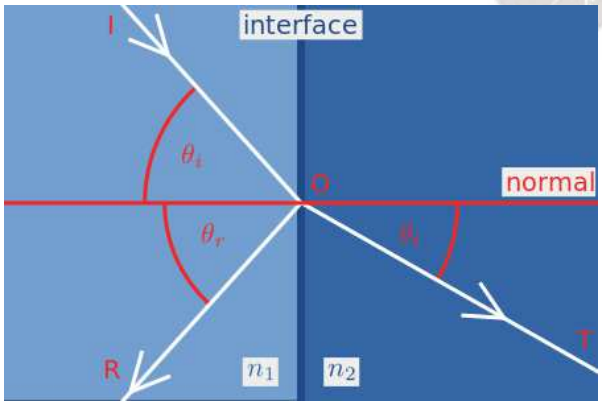


Refraction indices of some common materials

Vacuum	1.0
Water	1.333
Ice	1.309
Glass	1.5~1.6

Refraction Rays

- Behavior of light when moving between different refractive media
  - Fresnel Equation
  - Schlick's approximation to Fresnel equation



Reflection coefficient R

$$R(\theta) = R_0 + (1 - R_0)(1 - \cos\theta)^5$$

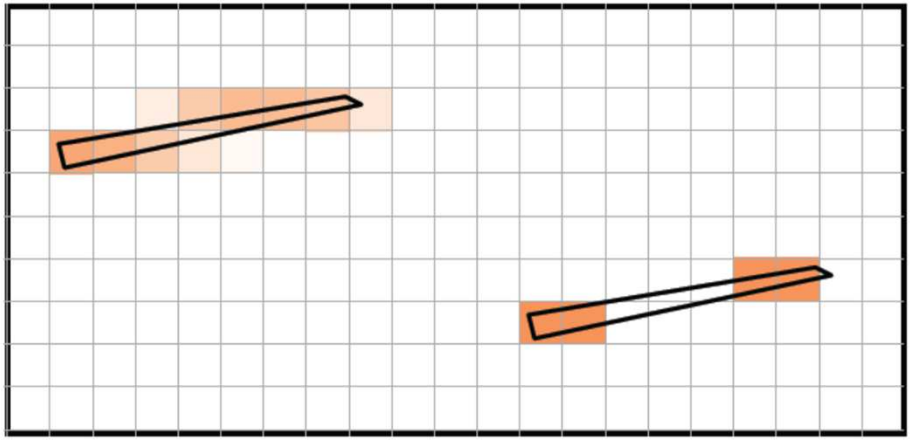
$$R_0 = \left(\frac{\eta_1 - \eta_2}{\eta_1 + \eta_2}\right)^2$$

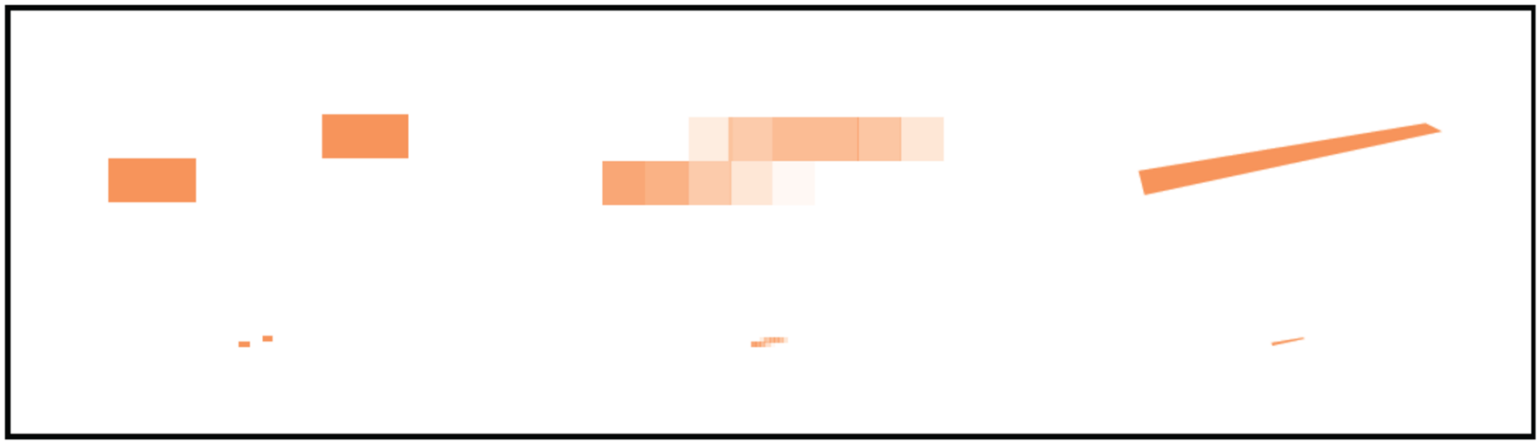
Transmission coefficient T

$$T(\theta) = 1 - R(\theta)$$

Aliasing Issue

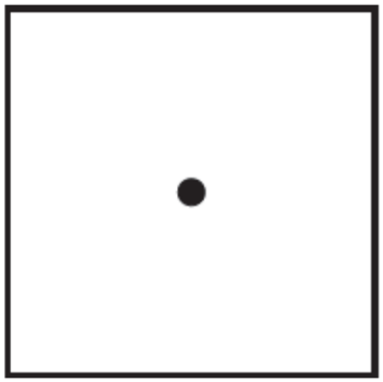
- Use only rays through pixel centers
- Need a solution of an integral over pixel



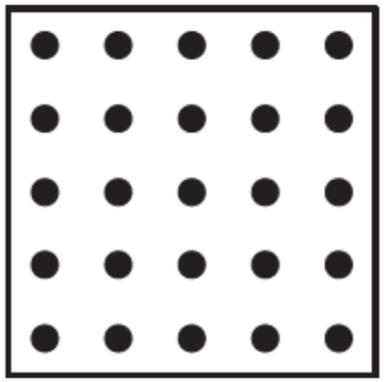


Distributed Ray Tracing

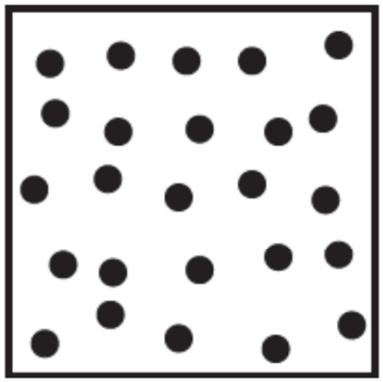
- Send multiple rays through each pixel



One Sample



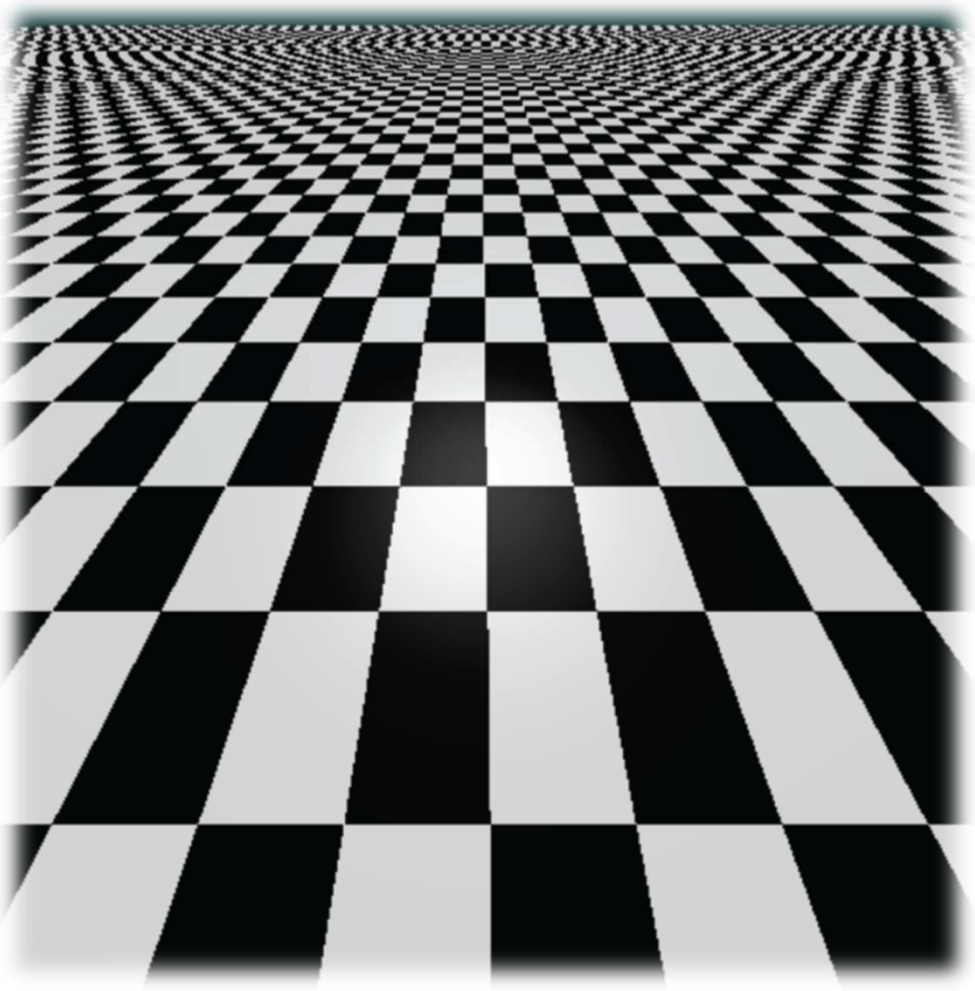
5x5 Grid



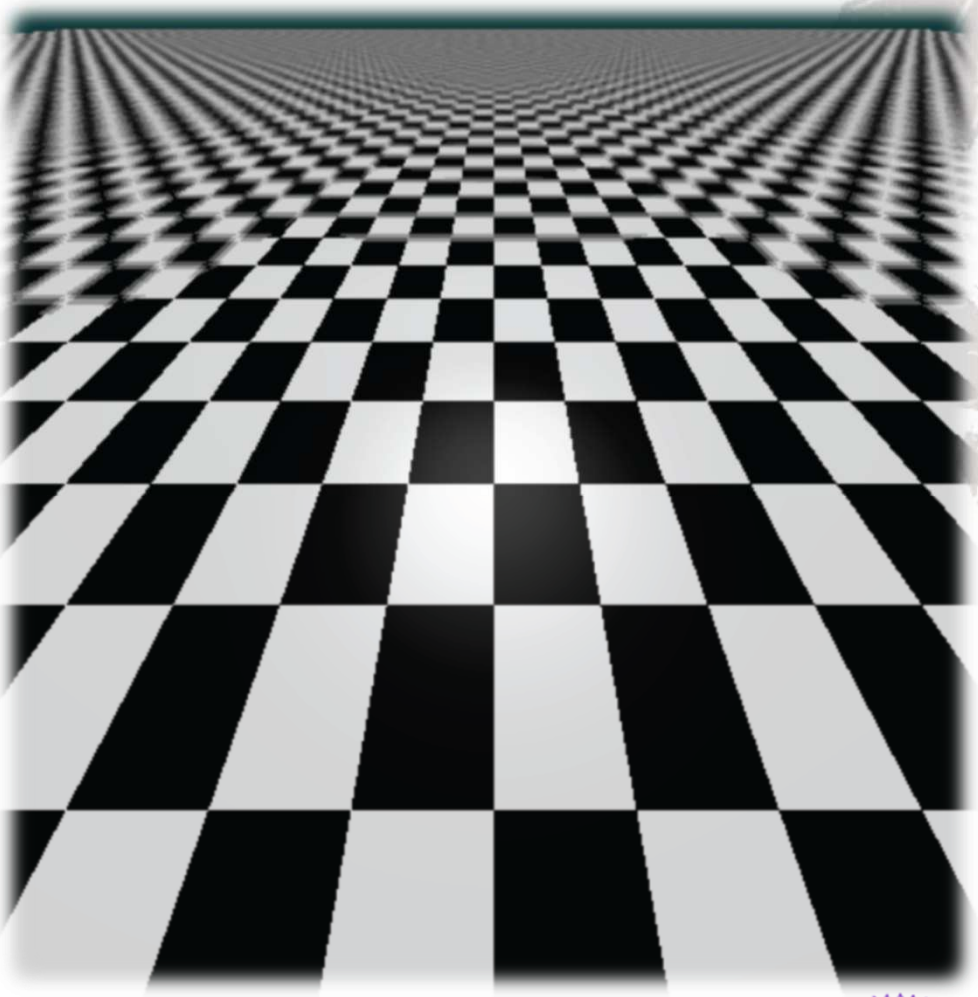
5x5 Jittered Grid

- Average results together

Distributed Ray Tracing



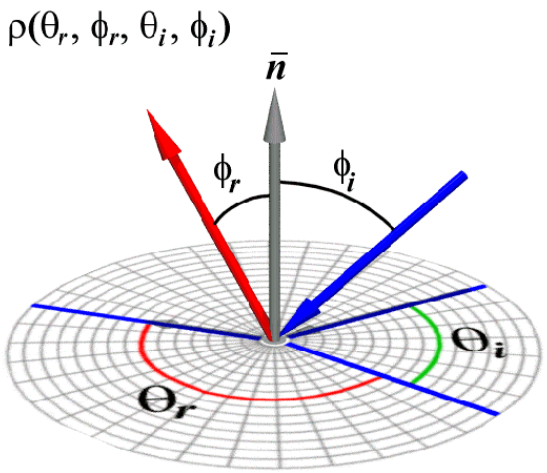
Without Distributed Ray Tracing



With Distributed Ray Tracing

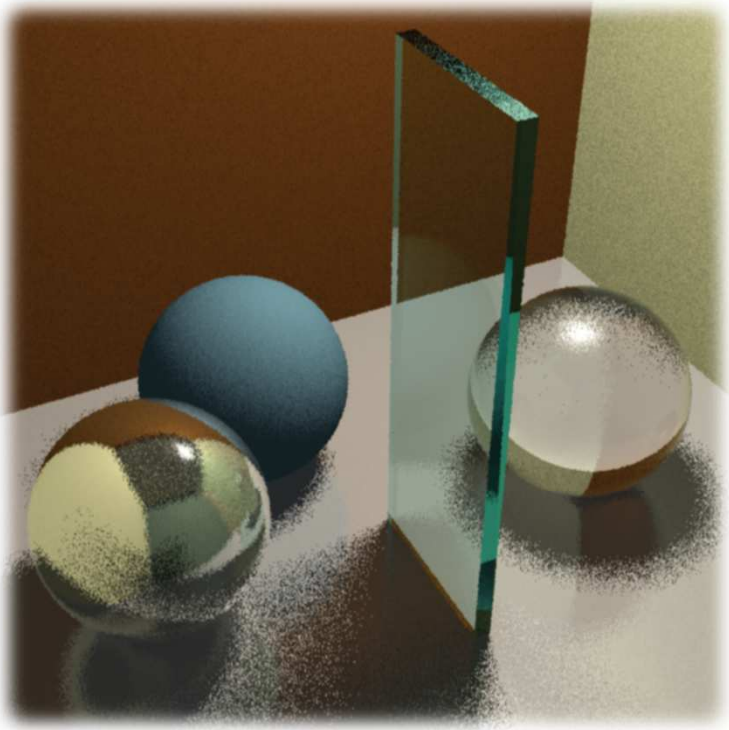
Distributed Ray Tracing

- Use multiple rays for reflection and refraction
  - At each bounce send out many extra rays
  - Quasi-random directions
  - Use BRDF (or Phong approximation) for weights

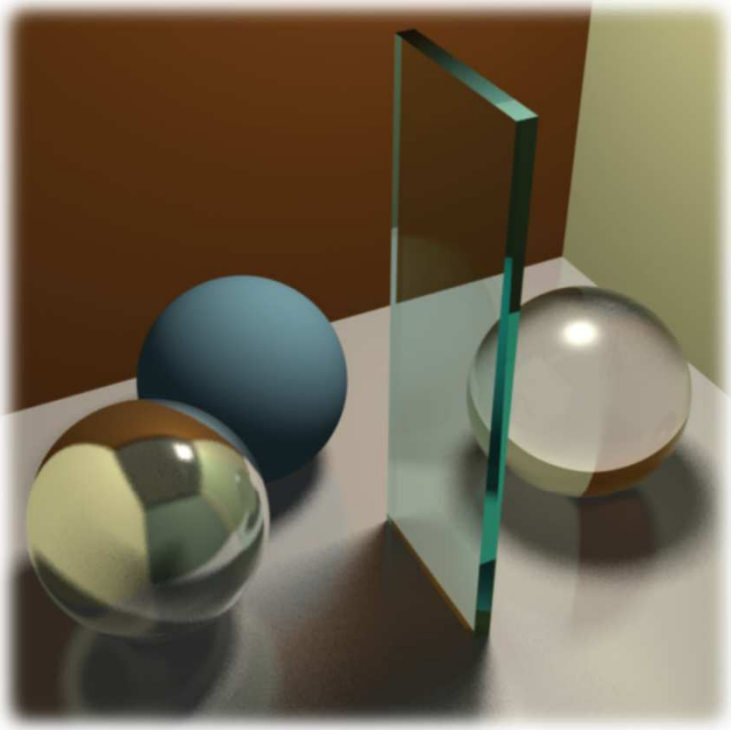


Distributed Ray Tracing

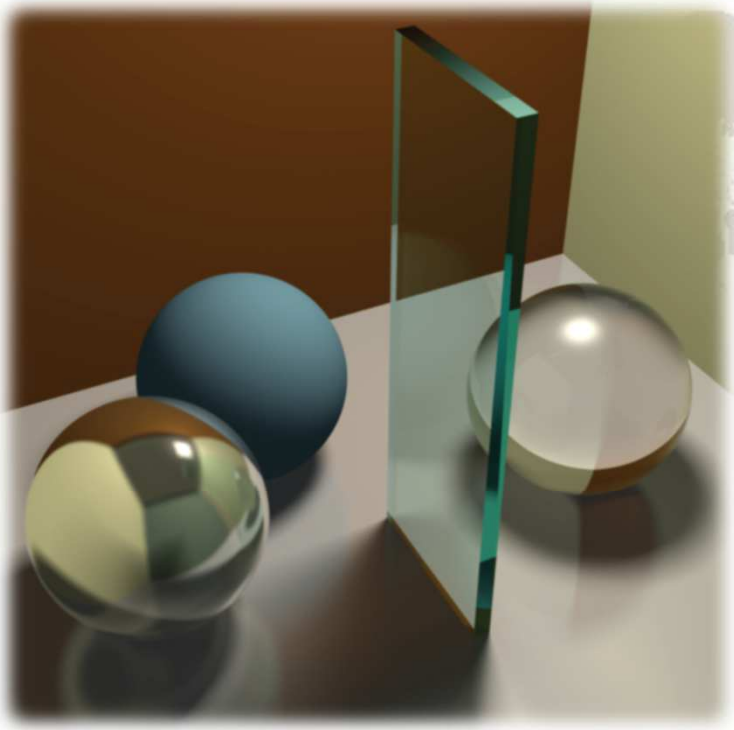
- How many rays?



1 sample per pixel



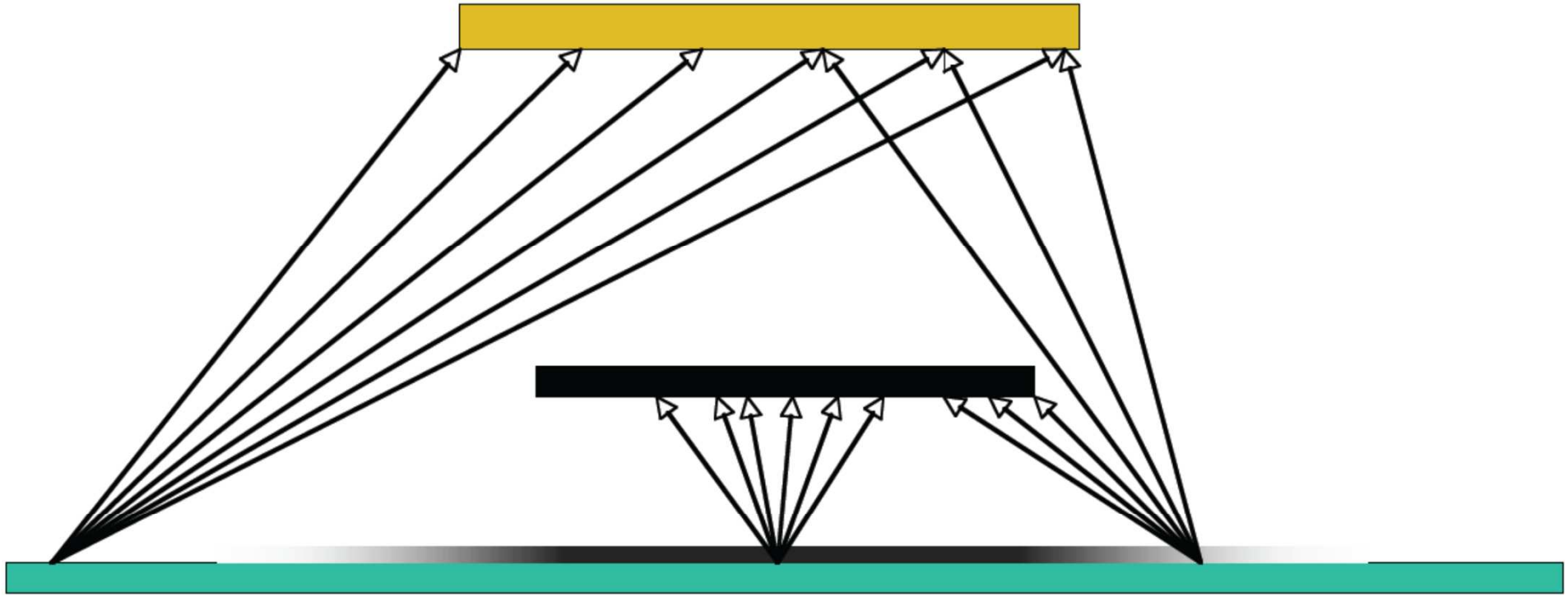
16 samples per pixel



256 samples per pixel

Soft Shadow

- Distribute shadow rays over light surface



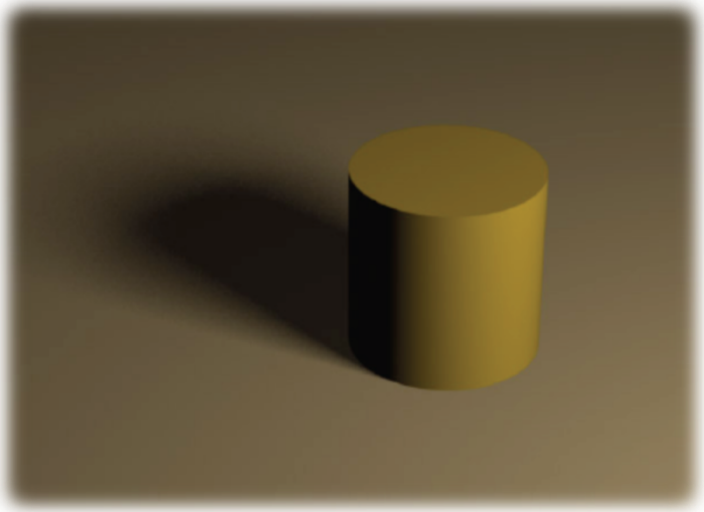
All shadow rays go through

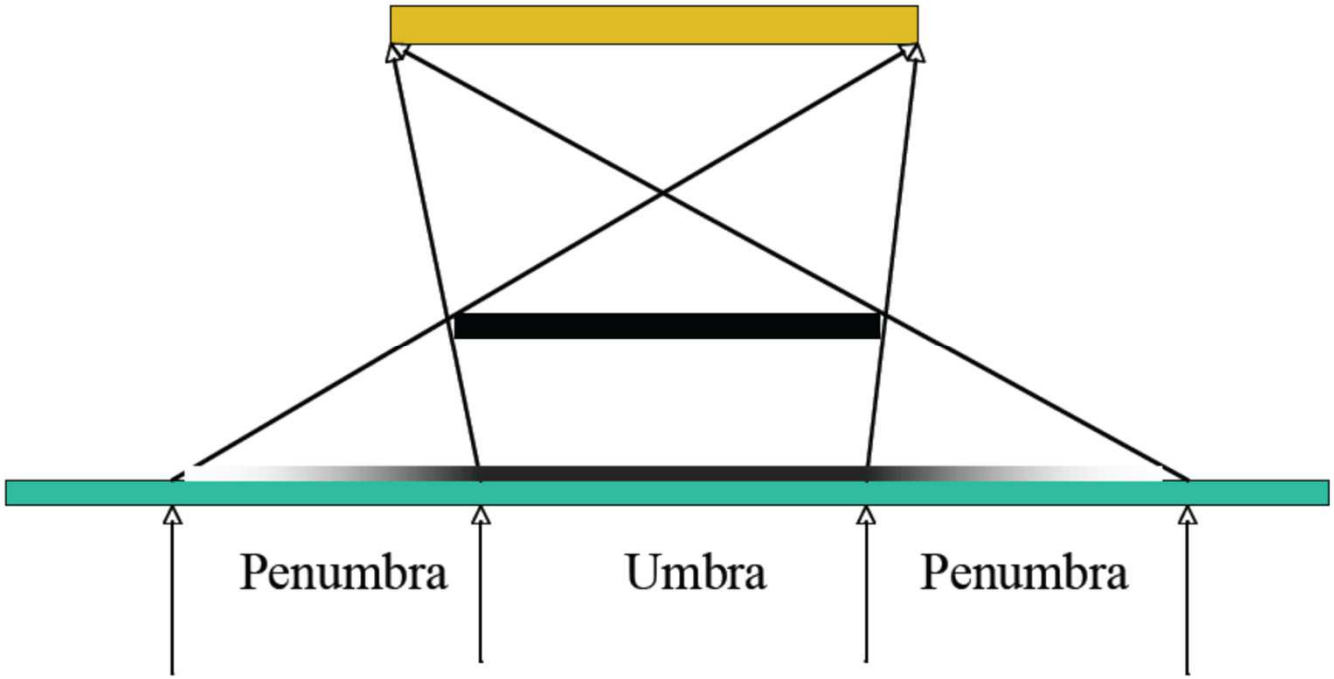
No shadow rays go through

Some shadow rays go through

Soft Shadow

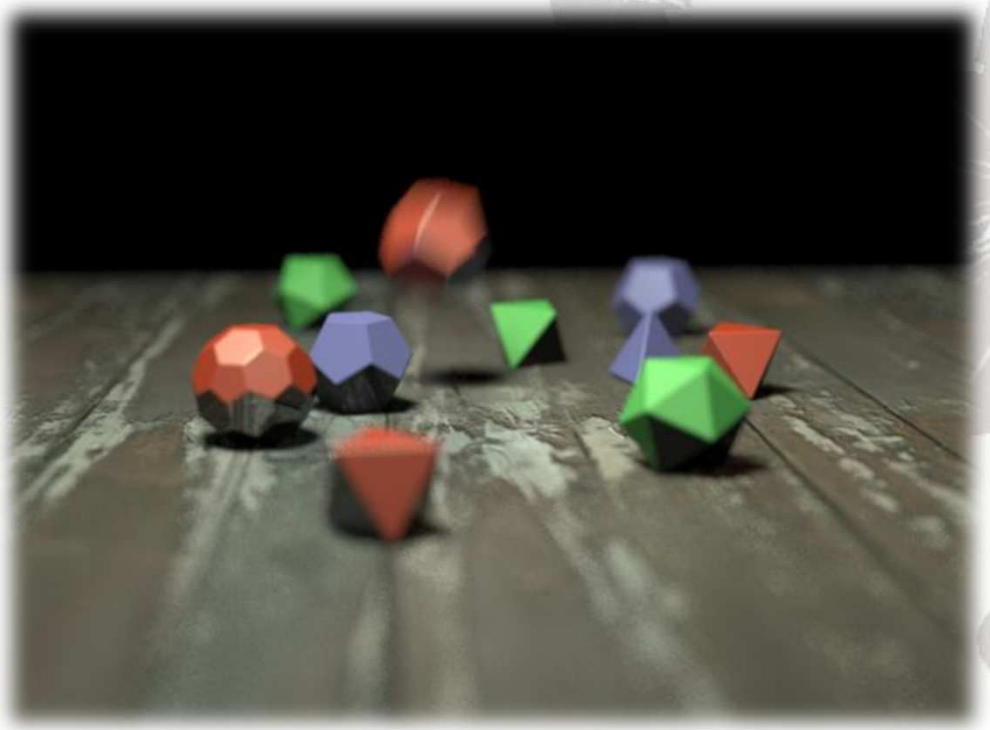
- Soft shadows result from non-point lights

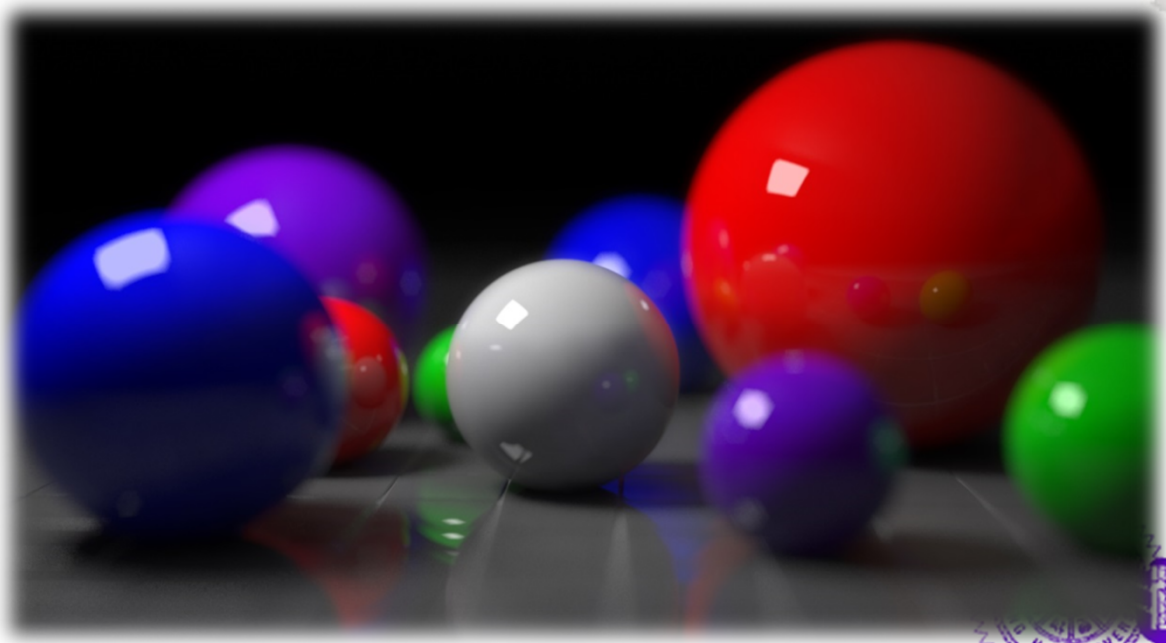




Other Effects

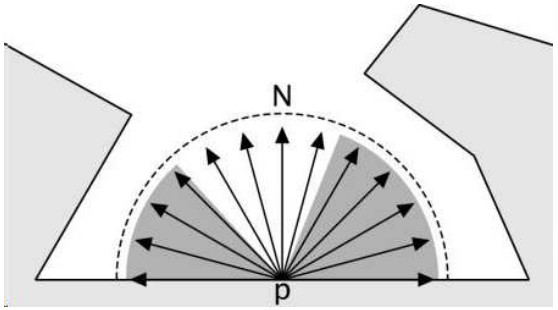
- Motion Blur
- Depth of Field

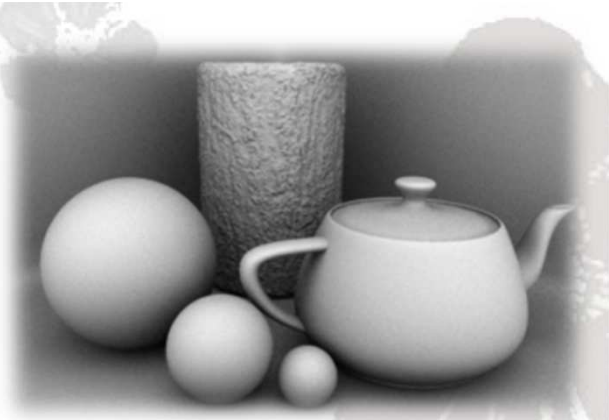




Other Effects

- **Ambient Occlusion**





Ambient Occlusion

Diffuse Lighting

Combined

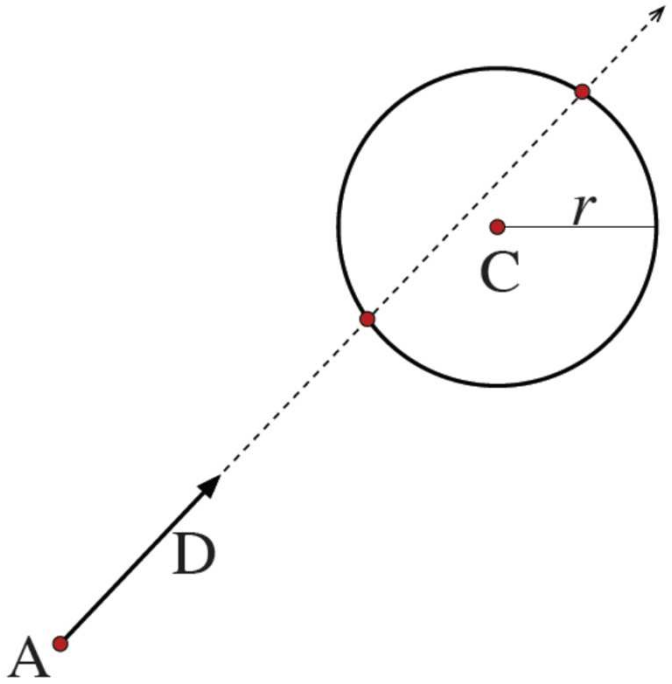
Intersection Test

- **Ray vs. Sphere**
  - **Ray equation:** $\mathbf{R}(t) = \mathbf{A} + t\mathbf{D}$
  - **Implicit equation for sphere:** $|\mathbf{X} - \mathbf{C}|^2 - r^2 = 0$
  - **Combine:**

    $$|\mathbf{R}(t) - \mathbf{C}|^2 - r^2 = 0$$

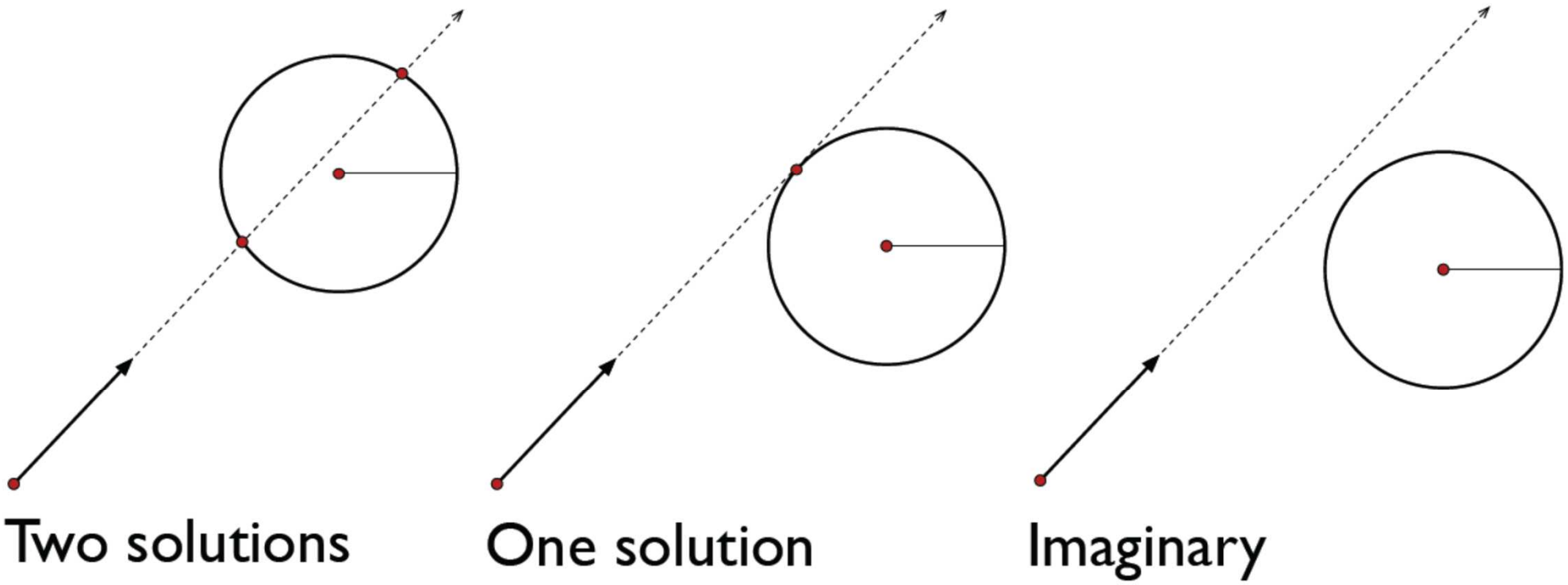
    $$|\mathbf{A} + t\mathbf{D} - \mathbf{C}|^2 - r^2 = 0$$

  - **Quadratic equation in** t



Intersection Test

- **Ray vs. Sphere**



Intersection Test

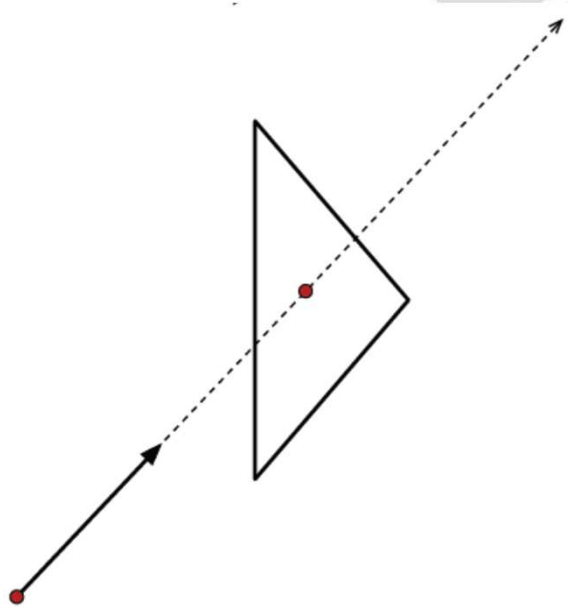
- **Ray vs. Triangle**
  - **Ray equation:** $\mathbf{R}(t) = \mathbf{A} + t\mathbf{D}$
  - **Triangle in barycentric coordinates:**

$$\mathbf{X}(\beta, \gamma) = \mathbf{V}_1 + \beta(\mathbf{V}_2 - \mathbf{V}_1) + \gamma(\mathbf{V}_3 - \mathbf{V}_1)$$

  - **Combine:**

$$\mathbf{V}_1 + \beta(\mathbf{V}_2 - \mathbf{V}_1) + \gamma(\mathbf{V}_3 - \mathbf{V}_1) = \mathbf{A} + t\mathbf{D}$$

  - **Solve for β, γ, and t**



Q&A