Global Illumination

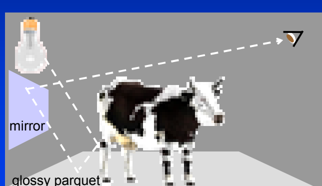
- Recursive ray tracing
- Radiosity
- Two-pass radiosity - ray tracing

Global Illumination Model

- Shading of a surface point is computed taking into account the other elements in the scene
- Light ray may hit several surfaces before it reaches the viewer: better approximation ↔ higher cost
- Produces more realistic effects

Light Transport

- Starting at a light source L, a light ray may bounce several times among specular (S) and diffuse (D) surfaces before reaching the viewer's eye (E)
- Regular Expression: L(S|D)*E, * means 0 or more times

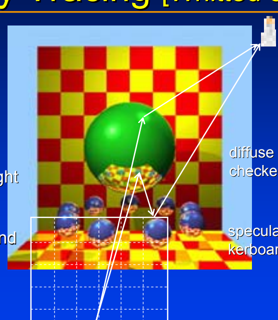


Global Illumination Algorithms

- Recursive Ray Tracing
  - Handles multiple inter-reflections among shiny surfaces, refraction and shadows (LS*E | LDS*E)
  - Produces high quality results for shiny surfaces
- Radiosity
  - Handles multiple inter-reflections among diffuse surfaces and color bleeding (LD*E)
  - Produces high quality results for diffuse environments
- Two-Pass Radiosity - Ray Tracing
  - Combines the strengths of both approaches (L(S|D)*E)

Recursive Ray Tracing [Whitted 80]

- Trace a ray from the eye through the center of each pixel
- Find the closest intersection along each ray
- At the closest intersection, shoot shadow rays (one per light source), locally evaluate an illumination function, and recursively span a reflection and refraction rays.
- Add the contributions of the spanned rays



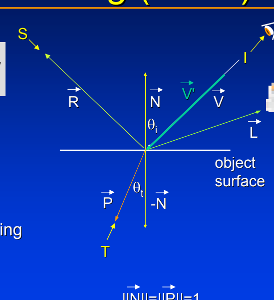
Recursive Ray Tracing (Cont.)

$$I = I_a + k_d \sum_{l=1}^{\#ligths} (\vec{N} \cdot \vec{L}_l) + k_s S + k_t T$$

S: intensity of incident light from direction R

T: intensity of incident light from direction P

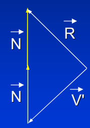
I: intensity of incident light reaching the viewer along direction V

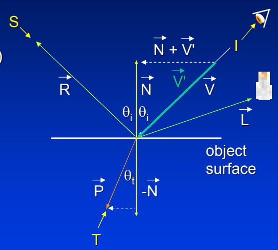


Recursive Ray Tracing (Cont.)

$\vec{V}' = \frac{\vec{V}}{|\vec{V} \cdot \vec{N}|}$ (cosα = - cos(180- α))

$\vec{R} = \vec{V}' + 2\vec{N}$





$\vec{P} = \sin\theta_t \frac{(\vec{N} + \vec{V}')}{|\vec{N} + \vec{V}'|} - \cos\theta_t \vec{N}$

Recursive Ray Tracing (Cont.)

According to Snell's law

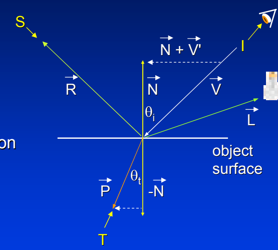
$\frac{\sin\theta_t}{\sin\theta_i} = \frac{\eta_{i\lambda}}{\eta_{t\lambda}}$

where η is the index of refraction

Thus,

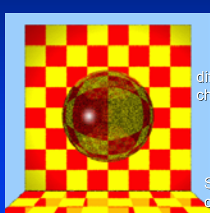
$\vec{P} = \sin\theta_t \frac{(\vec{N} + \vec{V}')}{|\vec{N} + \vec{V}'|} - \cos\theta_t \vec{N}$

$\vec{P} = \frac{\eta_{i\lambda}}{\eta_{t\lambda}} \sin\theta_i \frac{(\vec{N} + \vec{V}')}{|\vec{N} + \vec{V}'|} - \cos\theta_t \vec{N}$



Recursive Ray Tracing (Cont.)

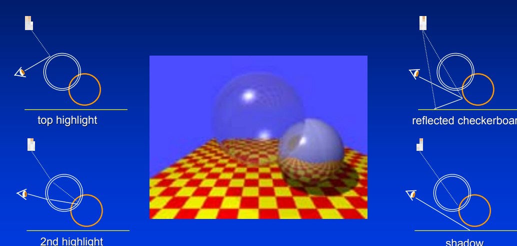
- Example: refraction produced by a glass sphere containing black particles



diffuse checkerboard

Specular checkerboard

Recursive Ray Tracing (Cont.)



Adapted from the book *3D Computer Graphics* by Alan Watt 3rd Edition, p. 348, Addison-Wesley. Whitted's classical ray tracing image rendered with software accompanying the book *The Computer Image* by Alan Watt and Fabio Policarpo, Addison-Wesley.

Recursive Ray Tracing (Cont.)

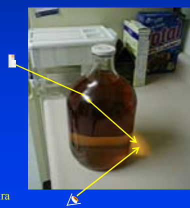
- Texture mapping objects in ray traced scenes

Two-Pass Ray Tracing [Arvo 86]

- Recursive ray tracing cannot easily handle caustics (LS*DE)

Two-Pass Ray Tracing (Cont.)

- 1st Pass:
  - trace rays from the light source through transparent and to specular objects until the ray reaches a diffuse surface and store the incident radiance in a light map associated with the surface
- 2nd Pass:
  - conventional ray tracing collecting the energy stored in the first pass



Radiosity Methods

- Radiosity: energy per unit area that leaves a surface
- Based on thermal-engineering models for emission and reflection of radiation
- Assume the conservation of light energy in a closed environment
- Determine light interaction among surfaces in a view-independent way
- Compute the radiosity (once) at the vertices of a mesh and interpolate the results using Gouraud shading

Radiosity Equation

- Discrete Equation

$$B_i = E_i + \rho_i \sum_{j=1}^{\#patches} B_j \underbrace{v(i,j)}_{\text{visibility term}} \underbrace{F_{i-j}}_{\text{form factor}}$$

B_i: Radiosity of patch *i*

E_i: Emissivity of patch *i*

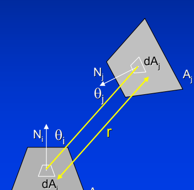
ρ_i: Reflectivity of patch *i*

v(i,j) = 1 (dA_j visible from dA_i); otherwise 0

form factor is a geometric term (does not depend on E_i, nor on ρ_i)

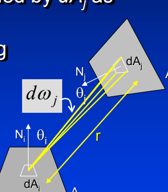
Form Factor

- Expresses the fraction of the energy leaving one surface and reaching the other (and vice-versa)
- It depends on the distance and orientation of the two surface elements



Form Factor Computation

- The fraction of light leaving dA_i and arriving at dA_j is proportional to the solid angle subtended by dA_j as viewed from dA_i
- The energy per unit solid angle leaving dA_i in a certain direction is constant per unit projected area
- Form factor is inversely proportional to the square distance and to $\cos\theta_j$



$$dF_{dA_i - dA_j} = \frac{\cos\theta_i}{\pi} d\omega_j = \frac{\cos\theta_i \cos\theta_j}{\pi r^2} dA_j$$

Form Factor Computation (Cont.)

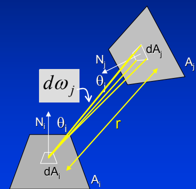
- Form factors involving finite areas can be obtained from

$$dF_{dA_i - dA_j} = \frac{\cos\theta_i \cos\theta_j}{\pi r^2} dA_j$$

- Thus

$$F_{dA_i - A_j} = \int_{A_j} \frac{\cos\theta_i \cos\theta_j}{\pi r^2} dA_j$$

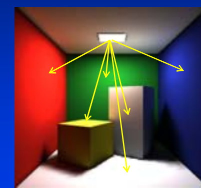
$$F_{A_i - A_j} = \frac{1}{A_i} \int_{A_i}\int_{A_j} \frac{\cos\theta_i \cos\theta_j}{\pi r^2} dA_j dA_i$$



Progressive Radiosity [Cohen 88]

- Provides immediate feedback by displaying the results after each iterative step

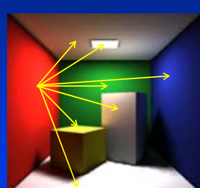
for (all i)
  $B_i = E_i$ ← unshot radiosity
  $\Delta B_i = E_i$
while (not congerged)
  pick i, such that $\Delta B_i * A_i$ is largest
  for (every element j)
    $\Delta rad = \Delta B_i * \rho_j F_{ji}$
    $\Delta B_j = \Delta B_j + \Delta rad$
    $B_j = B_j + \Delta rad$
  $\Delta B_i = 0$
  display the image



Progressive Radiosity [Cohen 88]

- Provides immediate feedback by displaying the results after each iterative step

for (all i)
  $B_i = E_i$ ← unshot radiosity
  $\Delta B_i = E_i$
while (not congerged)
  pick i, such that $\Delta B_i * A_i$ is largest
  for (every element j)
    $\Delta rad = \Delta B_i * \rho_j F_{ji}$
    $\Delta B_j = \Delta B_j + \Delta rad$
    $B_j = B_j + \Delta rad$
  $\Delta B_i = 0$
  display the image

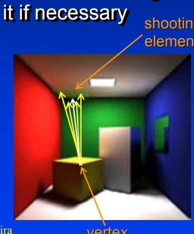


Ray Traced Form Factors [Wallace 89]

- Compute the radiosity directly for the vertices of the mesh
- For each vertex, trace shadow rays to the “shooting” element and adaptively subdivide it if necessary
- Form factor computed as

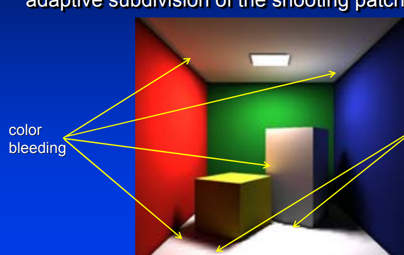
$$F_{dA_i - A_j} = \frac{A_j}{n} \sum_{k=1}^{n} \nu(i,k) \frac{\cos\theta_i \cos\theta_j}{\pi r^2 + (A_j / n)}$$

n is the number of sample points on Aj



Results

- Image rendered using ray traced form factors with adaptive subdivision of the shooting patches



Two-Pass Radiosity Ray-Tracing

- First Pass: view-independent radiosity solution using extended form factor to account for indirect visibility
- Second Pass: view-dependent solution (trace rays from the eye through each pixel covered by a specular surface

Multiple Mirrors

Two Mirrors

Three Mirrors