CS 775: Advanced Computer Graphics

Lecture 4: Radiosity
The Rendering Equation

\[
L_o(p, \omega_o) = L_e(p, \omega_o) + \int_{\Omega} f_r(p, \omega_o, \omega_i) L_i(p, \omega_i) \cos \theta_i \, d\omega_i
\]
The Rendering Equation

\[ L_o(p, \omega_o) = L_e(p, \omega_o) + \int_{\Omega} f_r(p, \omega_o, \omega_i) L_i(p, \omega_i) \cos \theta_i \, d\omega_i \]
The Rendering Equation

\[ L_o(x' \rightarrow x''') = L_e(x' \rightarrow x''') + \int_S f_r(x \rightarrow x' \rightarrow x''') L_i(x \rightarrow x') V(x', x') G(x, x') \, dA \]


\[ G(x, x') = \frac{\cos \theta_i \cos \theta_i'}{||x - x'||^2} \]
Solutions to the Rendering Eqn.

- OpenGL
- Ray Tracing
- Radiosity
- Distribution Ray Tracing and Path Tracing
- Photon Mapping
Light Paths

- A grammar for light paths
- Alphabet
  - L : Point on the light source
  - D : Point on a diffuse surface
  - S : Point on a specular surface
  - E : Point on the eye/camera
- Regex notation
  - ab – concatenate a AND b
  - a|b – either a OR b
  - a* – Zero or more repetition of a
  - a+ – One or more repetition of a
Light Paths

- All light paths: $L(D|S)^*E$

Rendering Trivia: The Cornell Box
http://www.graphics.cornell.edu/online/box
OpenGL

- LD|SE
- Point Lights
- Only Direct Illumination
- Lambertian/Phong BRDF
- Visibility ignored

\[
L_o(p, \omega_o) = L_e(p, \omega_o) + \int_{\Omega} f_r(p, \omega_o, \omega_i) L_i(p, \omega_i) \cos \theta_i \, d\omega_i
\]

\[
L_o(p, \omega_o) = L_a + L_e(p, \omega_o) + \sum_{1}^{nLights} f_r(p, \omega_o, \omega_i) L_i(p, \omega_i) \cos \theta_i
\]
Ray Tracing

\[ L_o(p, \omega_o) = L_e(p, \omega_o) + \int_{\Omega} f_r(p, \omega_o, \omega_i) L_i(p, \omega_i) \cos \theta_i \, d\omega_i \]

- LDS*E
- Specular Reflection and Transmission only
- No other Indirect Illumination
- Whitted – only point lights and hard shadows.

\[ L_o(p, \omega_o) = L_a + L_e(p, \omega_o) + \sum_{1}^{nLights} f_r(p, \omega_o, \omega_i) L_i(p, \omega_i) V(p, \omega_i) \cos \theta_i + \text{indirect specular} \]
Radiosity

\[ L_o(x' \rightarrow x'') = L_e(x' \rightarrow x'') + \int_S f_r(x \rightarrow x' \rightarrow x'') L_i(x \rightarrow x') V(x, x') G(x, x') dA \]

- LD*E
- Assume all surfaces are Lambertian
- The term has its origin in Thermodynamics
- Has the same dimensions/units as Irradiance/Radiant Emittance
Radiosity

\[ L_o(x' \rightarrow x'') = L_e(x' \rightarrow x'') + \int_S f_r(x \rightarrow x' \rightarrow x'') L_i(x \rightarrow x') V(x, x') G(x, x') dA \]

The All-Lambertian Assumption

\[ f_r(x \rightarrow x' \rightarrow x'') = k_d = \frac{\rho}{\pi} \]

\[ L_o(x') = L_e(x') + \frac{\rho}{\pi} \int_S L_i(x) V(x, x') G(x, x') dA \]

Convert to Radiosities

\[ B = \int_{\Omega} L \cos \theta d\omega \quad \text{gives} \quad L = \frac{B}{\pi} \]

\[ B_o(x') = B_e(x') + \frac{\rho}{\pi} \int_S B_i(x) V(x, x') G(x, x') dA \]
Radiosity

\[ B_o(x') = B_e(x') + \frac{\rho}{\pi} \int_S B_i(x) V(x, x') G(x, x') dA \]

Radiosity Approximation: Discretize the surface into smaller elements.

\[ B_i = E_i + \rho \sum_{j=1}^{N} B_j F_{ij} \]

where \[ F_{ij} = \frac{1}{A_i A_j} \int_{A_i} \int_{A_j} \frac{V_{ij} \cos \theta_i \cos \theta_o}{\pi r^2} dA_j dA_i \]
Radiosity

Form a system of Equations and Solve

\[ B_i = E_i + \rho_i \sum_{j=1}^{N} B_j F_{ij} \]
\[ B_i - \rho_i \sum_{j=1}^{N} B_j F_{ij} = E_i \]
\[ (1 - \rho_i F_{ii}) B_i - \rho_i \sum_{j=1, j \neq i}^{N} F_{ij} B_j = E_i \]

\[
\begin{bmatrix}
1 & -\rho_1 F_{1,1} & \cdots & \cdots & \cdots & \cdots & -\rho_1 F_{1,n} \\
-\rho_2 F_{2,1} & 1 & -\rho_2 F_{2,2} & \cdots & \cdots & \cdots & -\rho_2 F_{2,n} \\
\vdots & \vdots & \vdots & \ddots & \ddots & \ddots & \vdots \\
-\rho_{n-1} F_{n-1,1} & \cdots & \cdots & 1 & -\rho_{n-1} F_{n-1,1} & \cdots & -\rho_{n-1} F_{n-1,n} \\
-\rho_n F_{n,1} & \cdots & \cdots & \cdots & 1 & -\rho_n F_{n,1} & \cdots & \cdots & \cdots & \cdots & 1 & -\rho_n F_{n,1} \\
\end{bmatrix}
\begin{bmatrix}
B_1 \\
B_2 \\
\vdots \\
B_{n-1} \\
B_n \\
\end{bmatrix}
= 
\begin{bmatrix}
E_1 \\
E_2 \\
\vdots \\
E_{n-1} \\
E_n \\
\end{bmatrix}
\]
Radiosity

Intuition behind Radiosity is conservation of energy – How?

\[ B_i \, dA_i = E_i \, dA_i + \rho_i \int_{\Omega} B_j \, F_{ji} \, dA_j \]

\( F_{ji} \) or the Form Factor from \( j \) to \( i \) is the proportion of total power leaving patch \( j \) that is received by patch \( i \).

Since there is no loss of power in between,

\[ F_{ij} \, A_i = F_{ji} \, A_j \]

\[ B_i \, dA_i = E_i \, dA_i + \rho_i \int_{\Omega} B_j \, F_{ji} \, dA_j \]

or \( B_i \, dA_i = E_i \, dA_i + \rho_i \int_{\Omega} B_j \, F_{ij} \, dA_i \)

or \( B_i = E_i + \rho_i \int_{\Omega} B_j \, F_{ij} \)
Radiosity

$F_{ij}$ or the **Form Factor** is the proportion of total power leaving patch $i$ that is received by patch $j$.

Solid angle subtended by $dA_j$ as seen from $dA_i$ is

$$d\omega = \frac{dA_j \cos \theta_j}{r^2}$$

If exitant radiance leaving the surface at $dA_i$ in the direction $\theta_i$ is $dM_i$ in the solid angle $d\omega$.

Flux leaving $dA_i$ in the direction of $dA_j$ is given by

$$d\Phi = dM_i \cdot dA_i = L_i \cos \theta_i \cdot d\omega \cdot dA_i = \frac{L_i \cos \theta_i \cos \theta_j \cdot dA_j \cdot dA_i}{r^2} = \frac{M_i \cos \theta_i \cos \theta_j \cdot dA_j \cdot dA_i}{\pi r^2}$$

Total flux leaving $dA_i$ over the entire hemisphere is

$$\Phi = M_i \cdot dA_i$$

$$\Rightarrow F_{dA_i \to dA_j} = \frac{d\Phi}{\Phi} = \cos \theta_i \cos \theta_j \cdot dA_j$$
Radiosity

Deriving the Form Factor

$F_{ij}$ or the **Form Factor** is the proportion of total power leaving patch $i$ that is received by patch $j$.

Form factor from $dA_i$ to $dA_j$ is given by

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

Form factor from $dA_i$ to $A_j$ is given by

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

Form factor from $A_i$ to $A_j$ is given by

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

If distance $r$ is large compared to area of the two patches then

$$F_{ij} \approx F_{dA_i \rightarrow A_j} = \int_{A_j} \frac{\cos \theta_i \cos \theta_j dA_j}{\pi r^2}$$
Radiosity

Algorithm

1. Discretize the environment into patches.
2. Calculate the form-factor for every patch.
3. Solve the system of equations.
4. Render the scene with the computed radiosity.
Radiosity

- A **hemicube** is a half cube centered at the patch.

- The **Nusselt Analogue** Justification – Form factor of a patch is equivalent to the fraction of the unit circle that is formed by the projection of the patch.

- Form-factor of a patch is equal to sum of the form-factor of the pixels it covers.
Radiosity

**Hemicube Form-factors**

- **Advantages**
  - Pre-compute pixel form factors
  - Which pixels are covered can be obtained from projection on the corresponding hemicube faces planes.

For a pixel $q$ on the top surface of the hemicube,

$$\Delta F_q = \frac{1}{\pi (x^2 + y^2 + 1)^2} \Delta A$$

$$r = (x^2 + y^2 + 1)^{1/2}$$

$$\cos \theta_i = \cos \theta_j = \frac{1}{(x^2 + y^2 + 1)^{1/2}}$$
Radiosity

1. Discretize the environment into patches.
2. Calculate the form-factor for every patch.
3. **Solve the system of equations.**
4. Render the scene with the computed radiosity.
## Radiosity

### Gauss-Seidel Solution

\[
\begin{bmatrix}
1 - \rho_1 F_{1,1} & \cdots & \cdots & \cdots & -\rho_1 F_{1,n} \\
-\rho_2 F_{2,1} & 1 - \rho_2 F_{2,2} & \cdots & \cdots & -\rho_2 F_{2,n} \\
\vdots & \vdots & \ddots & \vdots \ & \vdots \\
-\rho_{n-1} F_{n-1,1} & \cdots & \cdots & 1 - \rho_{n-1} F_{n-1,n-1} & -\rho_{n-1} F_{n-1,n} \\
-\rho_n F_{n,1} & \cdots & \cdots & \cdots & 1 - \rho_n F_{n,n}
\end{bmatrix}
\begin{bmatrix}
B_1 \\
B_2 \\
\vdots \\
B_{n-1} \\
B_n
\end{bmatrix}
= 
\begin{bmatrix}
E_1 \\
E_2 \\
\vdots \\
E_{n-1} \\
E_n
\end{bmatrix}
\]

1. Compute Form-factors for all patches
2. for all \( i \) do \{ \( B_i = E_i \) \}
3. while (not converged) do
4. \{
5. for each \( i \) do
6. \{
7. \quad \text{sum} = 0;
8. \quad for all \( j \) except \( i \) do
9. \quad \quad \text{sum} += K_{ij} \cdot B_j;
10. \quad B_i = E_i - \text{sum};
11. \}\}
12. \}
Radiosity

Gauss-Seidel Solution

\[
\begin{bmatrix}
B_1 \\
B_2 \\
\vdots \\
B_{n-1} \\
B_n
\end{bmatrix} - \begin{bmatrix}
E_1 \\
E_2 \\
\vdots \\
E_{n-1} \\
E_n
\end{bmatrix} = \begin{bmatrix}
\rho_1 F_{1,1} & \cdots & \cdots & \cdots \\
-\rho_2 F_{2,1} & \rho_2 F_{2,2} & \cdots & \cdots \\
\vdots & \vdots & \ddots & \vdots \\
-\rho_{n-1} F_{n-1,1} & \cdots & \cdots & \rho_{n-1} F_{n-1,n-1} \\
-\rho_n F_{n,1} & \cdots & \cdots & \cdots & \rho_n F_{n,n}
\end{bmatrix} \begin{bmatrix}
B_1 \\
B_2 \\
\vdots \\
B_{n-1} \\
B_n
\end{bmatrix}
\]

Relaxing one row at a time

A Gathering Solution
Radiosity

A Shooting Solution

Progressive Solution

1. for all i do
2. {
3. \( B_i = E_i, \Delta B_i = E_i \)
4. Compute Form-factors \( F_{ij} \) if not done
5. }
6. while not converged do
7. {
8. Pick \( i \) such that \( \Delta B_i * A_i \) is largest
9. for every patch \( j \) except \( i \)
10. {
11. \( \Delta \text{rad} = \rho_j \Delta B_i A_i / A_j \)
12. \( \Delta B_j = \Delta B_j + \Delta \text{rad} \)
13. \( B_j = B_j + \Delta \text{rad} \)
14. }
15. \( \Delta B_i = 0 \)
16. }
Radiosity

\[
\begin{bmatrix}
B_1 \\
B_2 \\
\vdots \\
B_{n-1} \\
B_n
\end{bmatrix}
= 
\begin{bmatrix}
B_1 \\
B_2 \\
\vdots \\
B_{n-1} \\
B_n
\end{bmatrix}
+ 
\begin{bmatrix}
\rho_1 F_{1,1} \\
\rho_2 F_{2,1} \\
\vdots \\
\rho_{n-1} F_{n-1,1} \\
\rho_n F_{n,1}
\end{bmatrix}
\begin{bmatrix}
\rho_1 F_{1,n} \\
\rho_2 F_{2,n} \\
\vdots \\
\rho_{n-1} F_{n-1,n} \\
\rho_n F_{n,n}
\end{bmatrix}
\]

Evaluating a column at a time

A Shooting Solution
Radiosity

Algorithm

1. Discretize the environment into patches.
2. Calculate the form-factor for every patch.
3. Solve the system of equations.
4. Render the scene with the computed radiosity.
Radiosity

http://www.mpi-inf.mpg.de/resources/atrium/hab/chapter_3/chapter_3.html
Radiosity

- Adaptive meshing

Initial (uniform) meshing → Rough radiosity calculation

Polygon size evaluation

Final radiosity calculation ← Intensity-adaptive meshing

http://hawk.isc.chuo-u.ac.jp/makino-lab/person/itot/mytopics/radios.html
Radiosity

- Extentions
  - Discontinuity Meshing
  - Hierarchical Radiosity
  - Add Participating Media
  - Combine with RayTracing
Lambertian Surfaces

A **Lambertian** surface is one that follows **Lambert's law**: Illumination emitted by a surface in a particular direction varies as the cosine of the angle between the said direction and the normal to the surface.

\[ L_o = \frac{dM}{\cos \theta_o \, d\omega} = k \]

\[ M = \int_{H^2(\vec{n})} L_o \cos \theta_o \, d\omega = \pi L_o \]
A **Lambertian** surface is one that follows **Lambert's law**: Illumination received by a surface in a particular direction varies as the cosine of the angle between the said direction and the normal to the surface.

\[
L_i = \frac{dE}{\cos \theta_i \, d\omega} = k
\]

\[
E = \int_{H^2(\vec{n})} L_i \cos \theta_i \, d\theta_i = \pi L_i
\]
Lambertian Surfaces

A Lambertian surface is one that follows Lambert's law: Illumination reflected by a surface in a particular direction varies as the cosine of the angle between the said direction and the normal to the surface.

\[
L_o(p, \omega_o) = \int_{H^2(\vec{n})} f_r(p, \omega_o, \omega_i) dE(p, \omega_i) d\omega_i \\
= \int_{H^2(\vec{n})} f_r(p, \omega_o, \omega_i) L_i(p, \omega_i) \cos \theta_i d\omega_i = \rho L_i = \pi k_d L_i
\]

\[L_o = k = \pi k_d L_i\]