Radiosity

Classic radiosity = finite element method

Assumptions
  - Diffuse reflectance
  - Usually polygonal surfaces

Advantages
  - View independent solution

Techniques
  - Meshing
  - Form factors
  - Solving linear equations

First Radiosity Pictures ...

Parry Moon and Domina Spencer (MIT), Lighting Design, 1948
The Radiosity Equation

Assume diffuse reflection only
Solve for radiosity (2D function)

\[ B(x) = B_e(x) + \rho(x) E(x) \]

\[ B(x) = B_e(x) + \rho(x) \int_{m'} F(x,x') B(x') dA' \]

\[ F(x,x') = \frac{G(x,x')}{\pi} = \frac{\cos \theta \cos \theta'}{\pi \|x-x'\|^2} V(x,x') \]
Classic Radiosity Algorithm

- Mesh Surfaces into Elements
- Compute Form Factors Between Elements
- Solve Linear System for Radiosities
- Reconstruct and Display Solution

Simple Room Scene

Example from John Wallace
Derivation

Radiosity integral equation

\[ B(x) = B_e(x) + \rho(x) \int_{M^2} B(x') F(x, x') dA' \]

Piecewise constant basis functions

\[ B(x) = \sum_i B_i N_i(x) \]
\[ B_e(x) = \sum_i E_i N_i(x) \]
\[ \rho(x) = \sum_i \rho_i N_i(x) \]

Derivation

Convert integral equation to matrix equation

\[ B(x) = B_e(x) + \rho(x) \int_{M^2} F(x, x') \sum_j B_j N_j(x') dA' \]

\[ \sum_i B_i N_i(x) = \sum_i B_i N_i(x) + \sum_i \rho_i N_i(x) \left( \sum_j B_j \int_{M^2} F(x, x') N_j(x') dA' \right) \]

\[ \int \left( \sum_i B_i N_i(x) \right) = \sum_i B_i N_i(x) + \sum_i \rho_i N_i(x) \left( \sum_j B_j \int_{M^2} F(x, x') N_j(x') dA' \right) \] dA

\[ B_i A_i = E_i A_i + \rho_i \sum_j B_j \int_{M^2} \int_{M^2} F(x, x') N_i(x) N_j(x') dA dA' \]
Form Factor

Throughput

\[ T_{ij} = T_{ji} = \int_{A_i} \int_{A_j} \frac{\cos \theta'_o \cos \theta_i}{\pi \| x - x' \|^2} V(x, x') \, dA dA' \]

Reciprocity

\[ T_{ij} \equiv A F_{ij} \Rightarrow A F_{ij} = A F_{ji} \]

Summation

\[ \sum_j F_{ij} = \sum_i F_{ji} = 1 \]

Form factor is the percentage of light ...

Classic Radiosity

Power Balance

\[ B_i A_i = E_i A_i + \rho_i \sum_j B_j A_j F_{ji} \]

Reciprocity

\[ A_i F_{ij} = A_j F_{ji} \Rightarrow B_i = E_i + \rho_i \sum_j F_{ij} B_j \]

Radiosity System

\[
\begin{pmatrix}
1 - \rho_1 F_{i1} & -\rho_1 F_{i2} & \cdots & -\rho_1 F_{in} \\
-\rho_2 F_{21} & 1 - \rho_2 F_{22} & \cdots & -\rho_2 F_{2n} \\
\vdots & \vdots & \ddots & \vdots \\
-\rho_n F_{n1} & -\rho_n F_{n2} & \cdots & 1 - \rho_n F_{nn}
\end{pmatrix}
\begin{pmatrix}
B_1 \\
B_2 \\
\vdots \\
B_n
\end{pmatrix}
= \begin{pmatrix}
E_1 \\
E_2 \\
\vdots \\
E_n
\end{pmatrix}
\]
Form Factors

**Differential-differential - \( F(x,x') \, dA' \)**

\[
F_{dA,dA'} = \frac{\cos \theta_o' \cos \theta_i'}{\pi \| x - x' \|^2} \, V(x,x') \, dA'
\]

**Differential-finite - \( E_j(x) \)**

\[
F_{dA_i,A_j} = \int_{A_j} \frac{\cos \theta_o' \cos \theta_i}{\pi \| x - x' \|^2} \, V(x,x') \, dA'
\]

**Finite-finite - \( \Phi_{ij} \)**

\[
F_{A_i,A_j} = \frac{1}{A_j} \int_{A_j} \int_{A_i} \frac{\cos \theta_o' \cos \theta_i}{\pi \| x - x' \|^2} \, V(x,x') \, dA \, dA'
\]

---

**Analytical Form Factors**

\[
F_{A_1,A_2} = \frac{2}{\pi XY} \left\{ \ln \left[ \frac{(1 + X^2)(1 + Y^2)}{(1 + X^2 + Y^2)} \right]^{\frac{1}{2}} + X \sqrt{1 + Y^2} \, \tan^{-1} \frac{X}{\sqrt{1 + Y^2}} \right. \\
+ \left. Y \sqrt{1 + X^2} \, \tan^{-1} \frac{Y}{\sqrt{1 + X^2}} - X \tan^{-1} X - Y \tan^{-1} Y \right\}
\]

\[\begin{align*}
X &= \frac{a}{c} \\
Y &= \frac{b}{c}
\end{align*}\]
Hemicube Algorithm

First radiosity algorithm to deal with occlusion
1. Render scene from the point of view of each vertex/element
2. Compute delta form factors - contribution from each pixel

Typical resolution: 32x32

Hemicube Delta Form Factors

\[
F_{dA_i,A_j} = \sum_{p \in A_j} \Delta F_p
\]

\[
r = \sqrt{x^2 + y^2 + 1}
\]

\[
\cos \phi = \frac{1}{\sqrt{x^2 + y^2 + 1}}
\]

\[
\Delta F = \frac{\Delta A}{\pi (x^2 + y^2 + 1)^2}
\]

\[
r = \sqrt{1 + y^2 + z^2}
\]

\[
\cos \phi = \frac{1}{\sqrt{1 + y^2 + z^2}}
\]

\[
\Delta F = \frac{\Delta A}{\pi (1 + y^2 + z^2)^2}
\]
Hemicube Algorithms

Advantages
- First practical method -> Patent!
- Use existing rendering systems; Hardware!
- Computes all form factors in $O(n)$

Disadvantages
- Computes differential-finite form factor
- Aliasing errors due to sampling
  Randomly rotate/shear hemicube
- Proximity errors
- Visibility errors
- Expensive to compute a single form factor

Solve $[F][B] = [E]$

Direct methods: $O(n^3)$
- Gaussian elimination
  Goral, Torrance, Greenberg, Battaile, 1984

Iterative methods: $O(n^2)$

Convergence
- Energy conservation -> diagonally dominant -> converge

- Gauss-Seidel, Jacobi: Gathering
  Nishita, Nakamae, 1985
  Cohen, Greenberg, 1985

- Southwell: Shooting
  Cohen, Chen, Wallace, Greenberg, 1988
Iterative Solvers

Iteration

\[(I - F)^{-1} B = E\]
\[B = (I + B + B^2 + \cdots) E\]

\[B^0 = E\]
\[B^1 = E + FB^0\]
\[\ldots\]
\[B^n = E + FB^{n-1}\]

Gathering

Row of F times B
Calculate one row of F and discard

```
for(i=0; i<n; i++)
    B[i] = Be[i];

while( !converged ) {
    for(i=0; i<n; i++) {
        E[i] = 0;
        for(j=0; j<n; j++)
            E[i] += F[i][j] * B[j];
        B[i] = rho[i]*E[i];
    }
}
```
Shooting

for(i=0; i<n; i++)
B[i] = dB[i] = Be[i];
while( !converged ) {
  set i st dB[i] is the largest;
  for(j=0; j<n; j++)
    if(i!=j) {
      dBj = rho[j]*F[j][i]*dB[i];
      dB[j] += dBj;
      B[j] += dBj;
    }
  dB[i]=0;
}

Brightness order
Column of F times B

Results: Gathering vs. Shooting

From Cohen et al.

Figure 5.9: Convergence versus number of steps for three algorithms.
Accuracy

Reference Solution
Table in room sequence from Cohen and Wallace

Uniform Mesh

Artifacts

A. Blocky shadows
B. Missing features
C. Mach bands
D. Inappropriate shading discontinuities
E. Unresolved discontinuities

Error Image
Increasing Resolution

Adaptive Meshing
Megaradiosity

From S. Teller, T. Funkhouser, C. Fowler, P. Hanrahan