# Radiosity

Ron Goldman
Department of Computer Science
Rice University

# References

1. *Computer Graphics:  Principles and Practice* -- Foley and Van Dam Chapter 16, Section 13, Pages 793-806.

2. *Global Illumination* -- Sillion and Puech, Chapters 2–4.

3. *Examining Radiosity* -- Siggraph Course Notes 1993 <http://www.education.siggraph.org/slides/slides93/53_93.htm>

4. *Radiosity* -- Allen Martin http://www.cs.wpi.edu/~matt/courses/cs563/talks/radiosity.html

5. *Graphics Gems* I–V.

# Comparison

| *Ray Tracing* | *Radiosity* |
|---|---|
| Point Light Sources | Ambient Light |
| Diffuse and Specular Reflection | Diffuse Reflection |
| Sharp Shadows | Soft Shadows |
| Weak Model for Ambient Light | Models Emitters and Reflectors |

# Diffuse Reflection

*Lambert's Law* -- $I = I_p k_d \cos\theta$

- $I$ = reflected intensity
- $I_p$ = intensity of light source
- $k_d$ = diffuse reflection coefficient $0 \le k_d \le 1$
- $\theta$ = direction between surface normal ($N$) and vector to light source ($L$)
- $\cos\theta = N \bullet L$
  - $N$ = unit normal to surface
  - $L$ = direction to light source

*Observations*

- Diffuse reflection is the same in all directions
- Actually 3 equations -- one for each primary color (RGB).

# Diffuse Reflection

*Formula*

$$I = I_p k_d \cos(\theta) = I_p k_d (L \bullet N)$$

$I_p$ = intensity of point light source at infinity

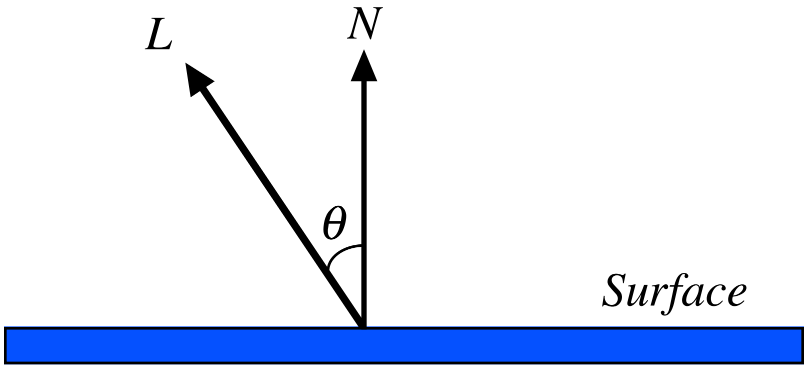
$k_d$ = diffuse reflection coefficient

$k_d$ is a material property

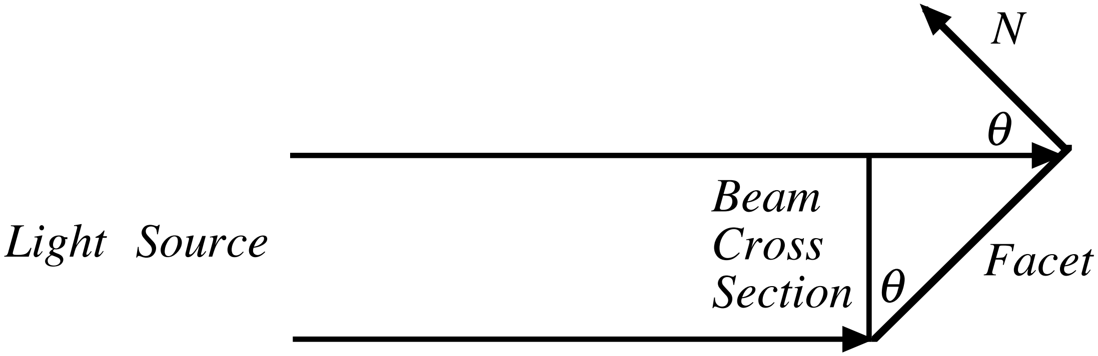
$0 \le k_d \le 1$

$\theta$ = angle between surface normal (*N*) and direction (*L*) to point light source

$\cos(\theta) = L \bullet N$



# Lambert's Law



$$I_{facet} = \frac{Light}{Unit\ \ Area} = \frac{Beam\ \ Cross\ \ Section}{Facet\ \ Area} \times I_{source}$$

$$\frac{Beam\ \ Cross\ \ Section}{Facet\ \ Area} = \cos(\theta)$$

$$I_{facet} = \cos(\theta) I_{source}$$

# Radiosity

- Radiosity = rate at which light energy leaves a surface = Power/Area
- Radiosity = emission + reflection
- Radiosity $\equiv$ Intensity($\times\pi$) {Intensity=Radiance=Power/Area/SAngle}
- Radiosity replaces: ambient intensity + diffuse intensity
- RADIOSITY IS VIEW INDEPENDENT
- View dependent calculations only for hidden surfaces
- Historically an Energy transfer method based on Heat transfer

# Rendering Equation

*Energy Conservation*

- Total Illumination = Emitted Energy + Reflected Energy
- $I(x,x') = E(x,x') + \int_S \rho(x,x',x'') I(x',x'') dx''$
    - $I(x,x')$ = energy passing from $x'$ to $x$
    - $E(x,x')$ = energy emitted directly from $x'$ to $x$
    - $\rho(x,x',x'')$ = reflection coefficient

*Remark*

- This is precisely the set up for recursive ray tracing!

# Radiosity Equation -- Continuous Form

*Radiosity = Emitted Energy + Reflected Energy*

$$B(x) = E(x) + \rho_d(x) \int_S B(y) \frac{\cos\theta \cos\theta'}{\pi r^2} V(x,y)\, dy$$

$B(x)$ = radiosity at $x$ -- identified with intensity (or energy)
(total power leaving a surface/unit area/solid angle)

$E(x)$ = energy emitted directly from a point $x$
(uniform in all directions -- diffuse emitter)

$\rho_d(x)$ = diffuse reflection coefficient -- $0 \le \rho_d(x) \le 1$

$V(x,y)$ = visibility term = 0 if $x$ not visible from $y$
= 1 if $x$ is visible from $y$

$\theta$ = angle between surface normal ($N$) at $x$ and (light) ray ($L$) to $y$

$\theta'$ = angle between surface normal ($N'$) at $y$ and (light) ray ($L$) to $x$

$r$ = distance from $x$ to $y$ -- inverse square law

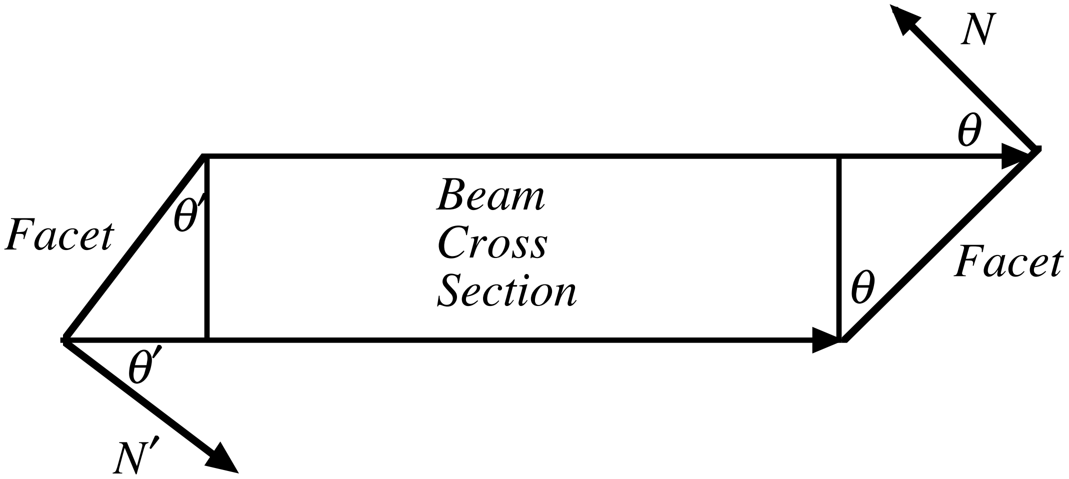
## Radiosity Equation -- Continuous Form (continued)

*Reflected Energy*

- $\int_S B(y)\frac{\cos\theta\cos\theta'}{\pi r^2}V(x,y)dy$ = energy reaching $x$ from all other $y$
- $\rho_d(x)\int_S B(y)\frac{\cos\theta\cos\theta'}{\pi r^2}V(x,y)dy$ = energy reflected from $x$

- Note $\cos\theta,\cos\theta'$ from Lambert's Law -- projection of flux onto surface.
- The factor $1/r^2$ comes from inverse square law
- Note mysterious factor of π in denominator.
  - Maps bidirectional flux to usual diffuse reflection coefficient.
    (See Sillion, page 14.)
  - Divide all reflection coefficients by π -- rescaling

# Lambert's Law



$$I_{facet} = \frac{Light\ Deposited}{Unit\ \ Area} = \frac{Beam\ \ Cross\ \ Section}{Facet\ \ Area} \times I_{source}$$

$$I_{source} = \frac{Light\ Emitted}{Unit\ \ Area} = \frac{Beam\ \ Cross\ \ Section}{Facet\ \ Area} \times I_{emitter}$$

$$\frac{Beam\ \ Cross\ \ Section}{Facet\ \ Area} = \cos(\theta), \cos(\theta')$$

$$I_{facet} = \cos(\theta)\cos(\theta')I_{emitter}$$

# The Radiosity Equation

*Radiosity Equation*

$$B(x) = E(x) + \rho_d(x) \int_S B(y) \frac{\cos\theta \cos\theta'}{\pi r^2} V(x,y) dy$$

*Problem*

- Difficult to solve Radiosity Equation because
  - Integral Equation
  - $B(u)$ appears on both sides of the equation!
  - Must know $B(u)$ to find $B(u)$!
- Actually 3 equations -- one for each primary color (RGB).

*Solution*

- Assume radiosity constant over a small patch
- Break the scene into small discrete patches

# Solving the Radiosity Equation

*Radiosity = Emitted Energy + Reflected Energy*

$$B(x) = E(x) + \rho_d(x) \int_S B(y) \frac{\cos\theta \cos\theta'}{\pi r^2} V(x,y)\, dy$$

$$B(x) = E(x) + \rho_d(x) \sum_{j=1}^{N} B_j \int_{P_j} \frac{\cos\theta \cos\theta'}{\pi r^2} V(x,y)\, dA_j \qquad (*)$$

*Get radiosity and energy of single patch as area weighted averages*

$$B_i = (1/A_i) \int_{P_i} B(x)\, dA_i$$

$$E_i = (1/A_i) \int_{P_i} E(x)\, dA_i$$

*Integrate discrete equation (*) to obtain*

$$B_i = E_i + \rho_i \sum_{j=1}^{N} B_j (1/A_i) \int_{P_i} \int_{P_j} \frac{\cos\theta \cos\theta'}{\pi r^2} V(x,y)\, dA_i\, dA_j$$

## Radiosity Equation -- Discrete Form

$$B_i = E_i + \rho_i \sum_{j=1}^{N} F_{ij} B_j \qquad i = 1, \ldots, N$$

$B_i$ = radiosity on patch $P_i$ -- identified with intensity

$E_i$ = energy emitted from patch $P_i$

(uniform in all directions -- diffuse emitter)

$F_{ij}$ = form factor -- depends only on geometry, color independent

$\rho_i$ = diffuse reflection coefficient for patch $P_i$ -- $0 \le \rho_i \le 1$

$\sum_{j=1}^{N} F_{ij} B_j$ = energy reaching patch $P_i$ from all other patches

$\rho_i \sum_{j=1}^{N} F_{ij} B_j$ = energy reflected from patch $P_i$

Actually 3 sets of equations -- one for each primary color (RGB).

# Form Factors

$F_{ij}$ = fraction of energy leaving patch $P_i$ arriving at patch $P_j$

**Theorem:** $F_{ij} = (1/A_i)\int_{P_i}\int_{P_j} \frac{\cos\theta\cos\theta'}{\pi r^2} V(x,y)\,dA_i\,dA_j$

Proof:

- $B_{ij}$ = radiosity transferred from $P_i$ to $P_j$ = $F_{ji}B_i$
- $B_i$ = energy radiated from $P_i$ per unit area
- $A_jB_{ij}$ = total energy transferred from $P_i$ to $P_j$ = $A_jF_{ji}B_i$

But

$$A_jF_{ji} = \int_{P_i}\int_{P_j} \frac{\cos\theta\cos\theta'}{\pi r^2(x,y)} V(x,y)\,dA_j\,dA_i = A_iF_{ij}.$$

Therefore

$$A_jB_{ij} = A_iB_iF_{ij}$$

so

$$F_{ij} = \frac{A_jB_{ij}}{A_iB_i} = \frac{\text{total energy transferred from } P_i \text{ to } P_j}{\text{the total energy radiated from } P_i}$$

# Properties of Form Factors

1. *Reciprocity*

$$A_i F_{ij} = A_j F_{ji} \qquad \left( = \int_{P_i} \int_{P_j} \frac{\cos\theta \cos\theta'}{\pi r^2} V(x,y)\, dA_i\, dA_j \right)$$

2. *Additivity*

$$F_{i,j \cup k} = F_{i,j} + F_{i,k}$$

3. *Partition of Unity (Conservation of Energy)*

$$\sum_j F_{ij} = 1$$

# Geometric Interpretation of Form Factors

(Look at 2-D projections first -- project onto a circle)

1. *Point Form Factor*

$$F_{ij} = (1/A_i)\int_{P_j}\int_{P_i}\frac{\cos\theta\cos\theta'}{\pi r^2}V(x,y)\,dA_i\,dA_j$$

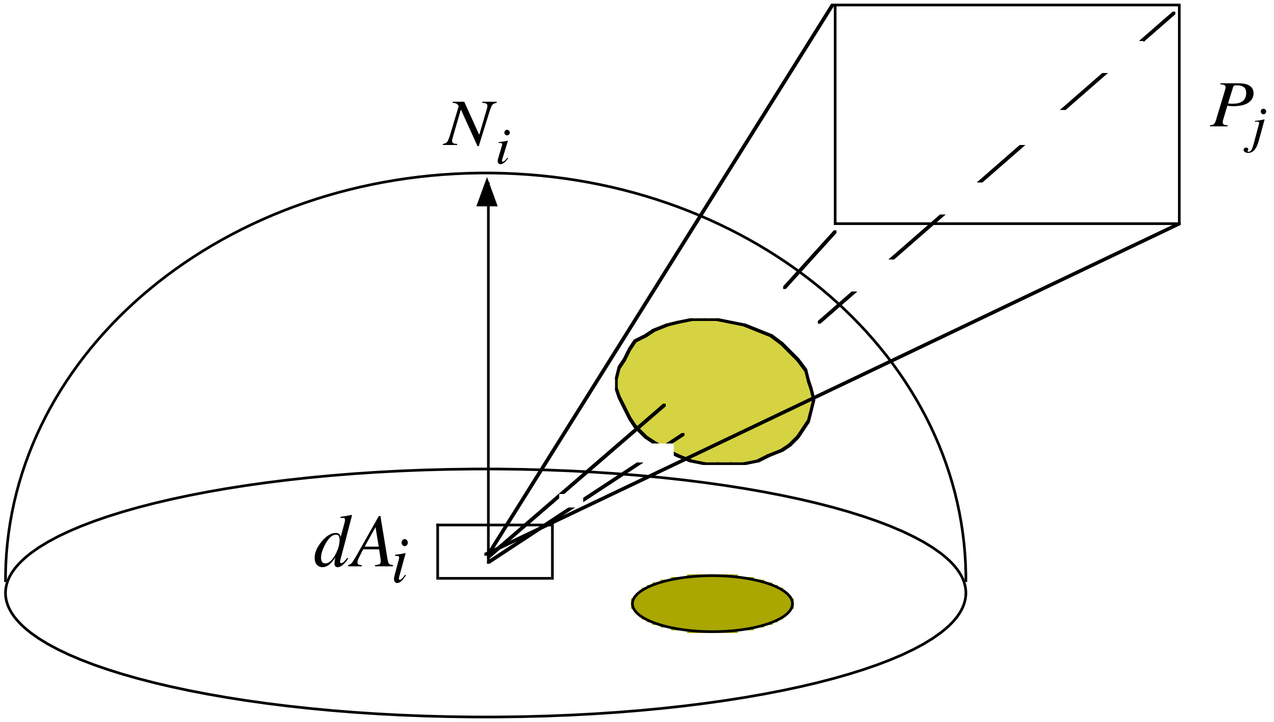
$$F_{di,j} = (1/dA_i)\int_{P_j}\int_{P_i}\frac{\cos\theta\cos\theta'}{\pi r^2}V(x,y)\,dA_i\,dA_j = \int_{P_j}\frac{\cos\theta\cos\theta'}{\pi r^2}V(x,y)\,dA_j$$

2. *Area Projections*
   - Onto Unit Hemi-Sphere -- $\frac{\cos\theta'}{r^2}$
   - Onto Base of Hemi-Sphere -- $\cos\theta$

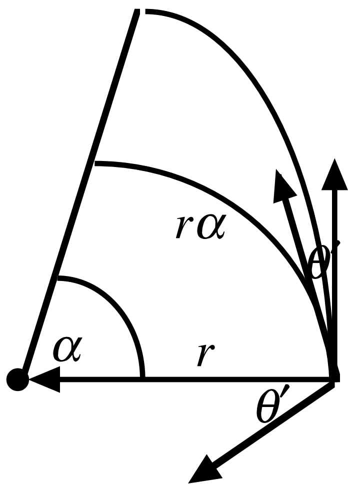
3. *Area of Base of Unit Hemi-Sphere = $\pi$*

4. *Note: Equal projections imply equal form factors*

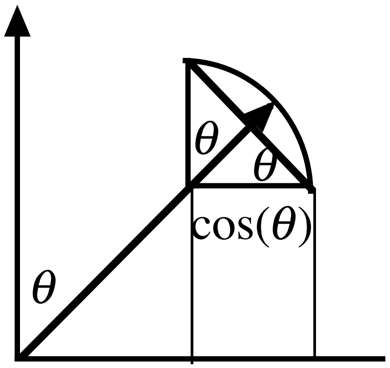
# Projections onto a Base of Hemisphere



# Projections onto a Circle



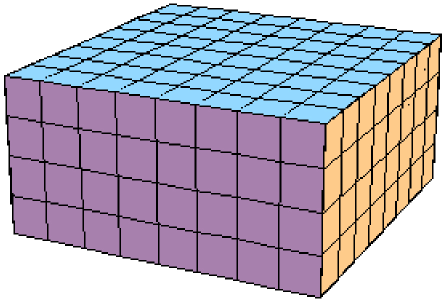
*Central Projection*
*Multiply by* $\cos(\theta')/r$



*Projection onto x–Axis*
*Multiply by* $\cos(\theta)$

# Hemi–Cubes

- Upper half unit cube (side=2) centered at a small patch $A_i$
- Projection on hemi-cube ≡ projection onto unit hemi-sphere
- Same projection ⇒ same form factor



# Computation of Form Factors

*Hemi-Cube Form Factor Algorithm*

- Compute form factors for each cell of the hemi-cube (below)
- Store these form factors
- Now find the form factor for every patch $P_j$ relative to $A_i$

For each face of the hemi-cube:

i. Clip the scene to the frustum determined by the center of $A_i$ and the face of the hemi-cube. {Pseudoperspective: frustum→box}

ii. For each cell of the hemi-cube face
Find the nearest polygon in the scene. {Apply a $z$-buffer algorithm.}
{Use orthogonal projection after, or perspective projection before, Pseudoperspective.}
Label each cell with the closest polygon

iii. For each polygon $P_j$
Sum the form factors of the cells labeled j

$$F_{di,j} = \sum_{q=j} \Delta F_q$$

# Observations

1. Use Reciprocity to Reduce the Amount of Computation
    - $A_i F_{ij} = A_j F_{ji}$

2. Recall the following algorithms
    - Perspective and Pseudoperspective
    - Clipping
    - Z-buffer

3. This computation of form factors takes most of the time.

# Form Factors for Hemi–Cube Cells

1. *Area of a Polygon*

$$Area(Polygon) = (1/2)\sum_{j=1}^{n}\left|P_j \times P_{j+1}\right|$$

2. *Point Form Factor*

$$F_{di,j} = \int_{P_j} \frac{\cos\theta\cos\theta'}{\pi r^2} V(x,y)\,dA_j$$

3. *Point to Cell Form Factor*

$$F_{di,dj} = \frac{\cos\theta\cos\theta'}{\pi r^2}\Delta A_j$$

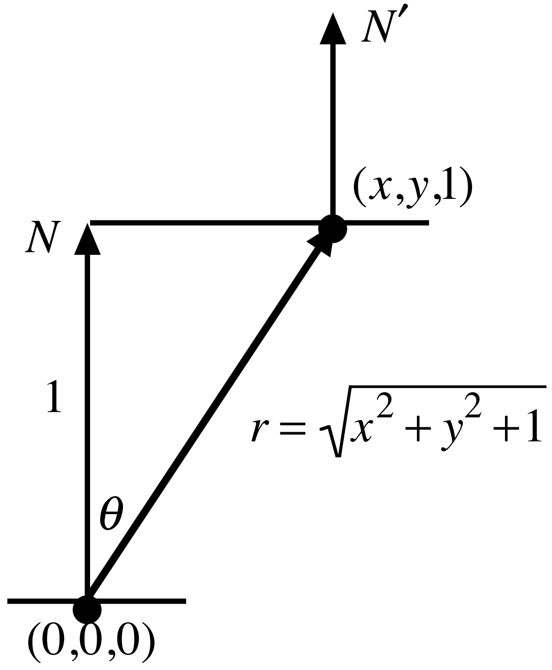
# Form Factors for Hemi–Cube Cells (continued)

4. *Top Face (Parallel to Surface -- 8 Fold Symmetry)*
   - $\cos\theta = \cos\theta' = 1/r$
   - $F_{di,dj} = \dfrac{\cos\theta\cos\theta'}{\pi r^2}\Delta A_j = \dfrac{\Delta A_j}{\pi r^4} = \dfrac{\Delta A_j}{\pi(x^2+y^2+1)^2}$

5. *Side Face (Parallel to y-axis at $x = \pm 1$ -- 4 Fold Symmetry)*
   - $\cos\theta = h/r = z/r \qquad \cos\theta' = 1/r$
   - $F_{di,dj} = \dfrac{\cos\theta\cos\theta'}{\pi r^2}\Delta A_j = \dfrac{h\Delta A_j}{\pi r^4} = \dfrac{z\Delta A_j}{\pi(y^2+z^2+1)^2}$

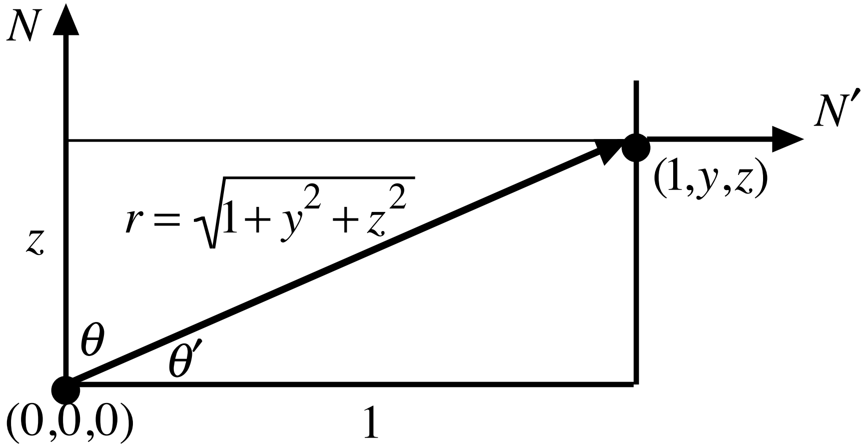
6. *Side Face (Parallel to x-axis at $y = \pm 1$ -- 4 Fold Symmetry)*
   - $\cos\theta = h/r = z/r \qquad \cos\theta' = 1/r$
   - $F_{di,dj} = \dfrac{\cos\theta\cos\theta'}{\pi r^2}\Delta A_j = \dfrac{h\Delta A_j}{\pi r^4} = \dfrac{z\Delta A_j}{\pi(x^2+z^2+1)^2}$

# Top Face Parallel to Surface



$$\cos\theta = \cos\theta' = 1/r$$

# Side Face Parallel to yz-Plane at $x = \pm 1$



$$\cos\theta = h / r = z / r \qquad \cos\theta' = 1 / r$$

## Recapitulation

*1. Mesh the Surfaces*

Break each surface into small surface patches

*2. Compute the Form Factors for each pair of surface patches*

Use the hemi-cube algorithm

*3*. Solve the linear system for the radiosities* $\{B_i\}$

$$B_i = E_i + \rho_i \sum_{j=1}^{N} F_{ij} B_j$$

Must do this computation 3 times, once for each primary (RGB)

4*. Compute radiosities at the vertices of the patches

Again do this computation 3 times, once for each primary (RGB)

## Recapitulation (continued)

5. Pick a Viewpoint

6. Determine which surfaces are visible
   Use any hidden surface algorithm

7. Apply Gouraud shading to the visible surfaces.

# Shading

*Flat Shading*

- Use Calculated Radiosity for Each Patch

*Gouraud Shading -- Regular Mesh*

- $I_{\text{int}} = (B_1 + B_2 + B_3 + B_4)/4$
- $I_{edge} = (B_1 + B_2)/2$
- $I_{corn} = B_1$

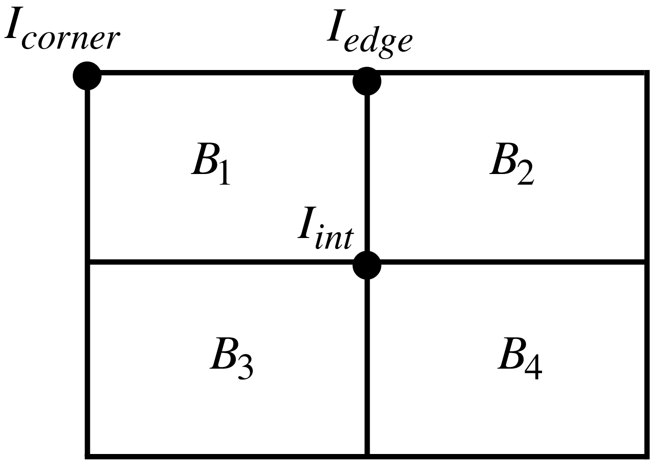
or

- $(I_{edge} + I_{\text{int}})/2 = (B_1 + B_2)/2$

  $\Rightarrow \quad I_{edge} = B_1 + B_2 - I_{\text{int}} = (3B_1 + 3B_2 - B_3 - B_4)/4$
- $(I_{corn} + I_{\text{int}})/2 = B_1$

  $\Rightarrow \quad I_{corn} = (7B_1 - B_2 - B_3 - B_4)/4$

# Gouraud Shading -- Regular Mesh



- $I_{\text{int}} = (B_1 + B_2 + B_3 + B_4)/4$
- $I_{edge} = (B_1 + B_2)/2$
- $I_{corn} = B_1$

or

- $(I_{edge} + I_{\text{int}})/2 = (B_1 + B_2)/2$

$\Rightarrow \quad I_{edge} = B_1 + B_2 - I_{\text{int}} = (3B_1 + 3B_2 - B_3 - B_4)/4$

- $(I_{corn} + I_{\text{int}})/2 = B_1$

$\Rightarrow \quad I_{corn} = (7B_1 - B_2 - B_3 - B_4)/4$

## The Radiosity System of Linear Equations (*N* Large)

*Individual Equations*

$$B_i = E_i + \rho_i \sum_{j=1}^{n} F_{ij} B_j \Leftrightarrow \sum_{j=1}^{n} (\delta_{ij} - \rho_i F_{ij}) B_j = E_i$$

*System of Equations*

$$\begin{aligned}
(1-\rho_1 F_{11})B_1 - \rho_1 F_{12} B_2 - \cdots - \rho_1 F_{1n} B_n &= E_1 \\
-\rho_2 F_{21} B_1 + (1-\rho_2 F_{22})B_2 - \cdots - \rho_2 F_{2n} B_n &= E_2 \\
\vdots \qquad\qquad & \quad \vdots \\
-\rho_n F_{n1} B_1 - \rho_n F_{n2} B_2 - \cdots + (1-\rho_n F_{nn})B_n &= E_n
\end{aligned}$$

*Matrix Form:* $M * B = E$

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

# Dictionary

$B_i$ = Radiosity on patch $P_i$ -- identified with intensity -- UNKNOWNS

$E_i$ = Energy emitted from patch $P_i$ -- USER SPECIFIED
(Uniform in all directions -- Diffuse emitter)

$F_{ij}$ = Form factor -- depends only on geometry -- COMPUTED

$\rho_i$ = Diffuse reflection coefficient for patch $P_i$ -- $0 \le \rho_i \le 1$ -- USER SPECIFIED

# Gathering

*Equations*

$$M * B = E$$

$$\sum_{j=1}^{n} M_{ij} B_j = E_i \quad \Rightarrow \quad B_i = \frac{E_i}{M_{ii}} - \sum_{j \neq i} \frac{M_{ij}}{M_{ii}} B_j \qquad (i^{th} \ \ row)$$

*Initial Guess*

$$B^0 = \begin{pmatrix} B_1^0 \\ B_2^0 \\ \vdots \\ B_n^0 \end{pmatrix} = initial \ \ guess. \qquad Try \ \ B^0 = \begin{pmatrix} E_1 \\ E_2 \\ \vdots \\ E_n \end{pmatrix} \ \ or \ \ B^0 = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}.$$

# Gathering (continued)

*Jacobi Relaxation*

$$B_i^p = \frac{E_i}{M_{ii}} - \sum_{j \neq i} \frac{M_{ij}}{M_{ii}} B_j^{p-1} \qquad 1 \le i \le N$$

*Gauss-Seidel Relaxation*

$$B_i^p = \frac{E_i}{M_{ii}} - \sum_{j=1}^{i-1} \frac{M_{ij}}{M_{ii}} B_j^p - \sum_{j=i+1}^{N} \frac{M_{ij}}{M_{ii}} B_j^{p-1} \qquad 1 \le i \le N$$

# Convergence

*Diagonally Dominant*

- Both methods converge for any initial guess when

  $|M_{ii}| \geq \sum_{j \neq i} |M_{ij}|$ (diagonally dominant).

- True for us, since form factors sum to 1, and $0 \leq \rho_i < 1$.

  $$Ax = b \Leftrightarrow Qx = (Q - A)x + b \Leftrightarrow x = (I - Q^{-1}A)x + Q^{-1}b$$

  $$x = fixed \;\; point \;\; of \;\; T(x) = (I - Q^{-1}A)x + Q^{-1}b$$

*Observation*

- $T$ is a contractive map when
  - $A$ is diagonally dominant
  - Q is diagonal part of $A$ (*Jacobi*)
  - $Q$ is the lower triangular part of $A$ *(Gauss-Seidel)*

# Shooting -- Progressive Refinement

*Radiosity Equation*

- $B_i = E_i + \sum_{j=1}^{N} \rho_i F_{ij} B_j$

*Gathering*

- $B_i$ *due to* $B_j = \rho_i F_{ij} B_j$
- Needs all the form factors for every patch.
- One hemicube → one patch radiosity

*Shooting*

- $A_i F_{ij} = A_j F_{ji}$ (Reciprocity)
- $B_j$ *due to* $B_i = \rho_j F_{ji} B_i = \rho_j F_{ij} B_i \frac{A_i}{A_j}$
- Need only form factors $\{F_{ij}\}$ for patch $P_i$ to update all patches $P_j$!
- One hemi-cube at a time.
- One hemicube → many patch radiosities

## Shooting (continued)

*Updates*

- $\Delta B_j = (\Delta B_j)_{old} + \rho_j F_{ij} \Delta B_i \dfrac{A_i}{A_j}$
- $B_j = (B_j)_{old} + \rho_j F_{ij} \Delta B_i \dfrac{A_i}{A_j}$
- $\Delta B_i = 0 \qquad (shooting\ \ surface)$

# **Shooting Algorithm**

a. Initialization

$$B^0 = \begin{pmatrix} E_1 \\ E_2 \\ \vdots \\ E_n \end{pmatrix} = \Delta B^0$$

b. Select patch $P_i$ for which the total power $A_i \Delta B_i$ is maximal.

c. Compute the form factors $\{F_{ij}\}$ for $P_i$, using one hemi-cube.

d. Update $B_j$ and $\Delta B_j$ for all patches $P_j$. (Shooting)

e. Display scene using current radiosity values $B_j$.

f. Go to b.

# **Ambient Correction**

*Average Diffuse Reflection*

$$\rho_{av} = \sum_{i=1}^{n} \rho_i A_i \,/\, \sum_{i=1}^{n} A_i$$

*Average Reflection*

$$R = 1 + \rho_{av} + \rho_{av}^2 + \cdots = \frac{1}{1 - \rho_{av}}$$

*Average Unshot Radiosity*

$$\Delta B_{av} = \sum_{i=1}^{n} \Delta B_i A_i \,/\, \sum_{i=1}^{n} A_i$$

*Ambient (Unshot) Light*

$$A = R\,\Delta B_{av}$$

*Ambient Correction (for display only)*

$$B_i = B_i + \rho_i A$$

# Summary

*Advantages*

- Photorealism
- Energy Transfer
- Soft Shadows
- Color Bleeding
- View Independence

*Disadvantages*

- Expensive -- Time and Space
- Must Approximate Curved Surfaces with Polygons (Meshing)
- No Specular Reflections

# Enhancements

1. Specular Reflection -- 2 pass algorithm
2. Participating Medium (atmosphere, fog) -- volumes
3. Realistic Light Sources (Sillion p.215)
4. Adaptive Mesh Generation (More Accurate Form Factors -- p. 799)
5. Monte Carlo Integration (ray tracing form factor computation)
6. More Accurate Approximations to Radiosity (not constant on patch)
    - Finite Element Methods
    - Wavelets

# Light Sources

*Emitters not Reflectors*

- $E > 0 \qquad \rho = 0$

*Form Factors*

- $F_{ij} = (1/A_i)\int_{P_i}\int_{P_j} \frac{S(\theta)\cos\theta'}{r^2} V(x,y)\,dA_i\,dA_j$ (arbitrary source)

- $F_{ij} = \int_{P_j} \frac{S(\theta)\cos\theta'}{r^2} V(x,y)\,dA_j$ (point source)

$$S(\theta) = light\ \ distribution$$

$$\int_{\Omega} S(\theta)\,d\omega = 1$$

*Sample Distributions*

- $S(\theta) = \cos\theta/\pi$ -- Diffuse
- $S(\theta) = \frac{\cos^n\theta}{n+1}$ -- Spotlight
- $S(\theta) = 1/4\pi$ -- Isotropic