

Radiosity

Measures of Illumination
The Radiosity Equation
Form Factors
Radiosity Algorithms

Alternative Notes

- SIGGRAPH 1993 Education Slide Set – Radiosity Overview, by Stephen Spencer

www.siggraph.org/education/materials/HyperGraph/radiosity/overview_1.htm

Limitations of Ray Tracing

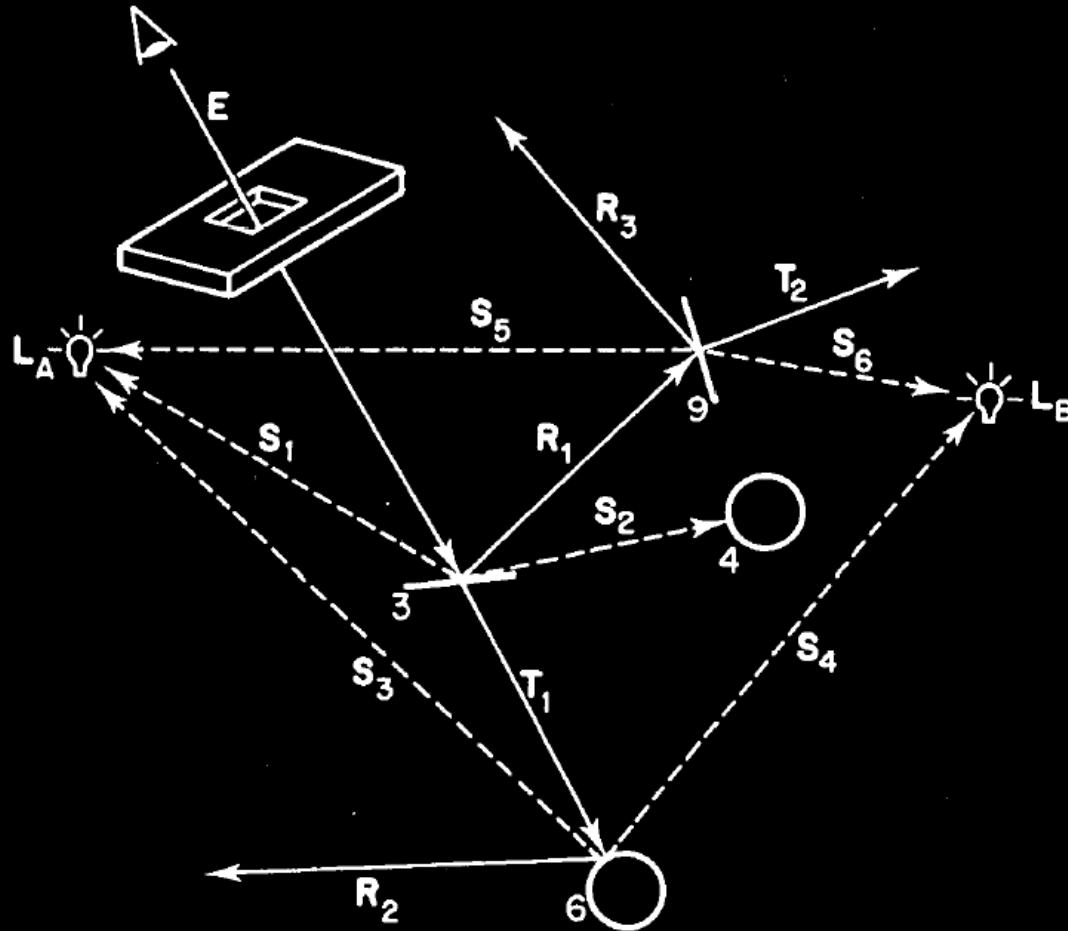


Fig. 11. An eye ray E propagated through a scene. Many of the intersections spawn reflected, transmitted, and shadow rays.

Local vs. Global Illumination

- Local illumination: **Phong model** (OpenGL)
 - Light to surface to viewer
 - No shadows, interreflections
 - Fast enough for interactive graphics
- Global illumination: **Ray tracing**
 - Multiple specular reflections and transmissions
 - Only one step of diffuse reflection
- Global illumination: **Radiosity**
 - All diffuse interreflections; shadows
 - Advanced: combine with specular reflection

Image vs. Object Space

- Image space: **Ray tracing**
 - Trace backwards from viewer
 - View-dependent calculation
 - Result: rasterized image (pixel by pixel)
- Object space: **Radiosity**
 - Assume only diffuse-diffuse interactions
 - View-independent calculation
 - Result: 3D model, color for each surface patch
 - Can render with OpenGL

Classical Radiosity Method

- Divide surfaces into patches (**elements**)
- Model light transfer between patches as system of linear equations
- Important assumptions:
 - Reflection and emission are **diffuse**
 - Recall: diffuse reflection is equal in all directions
 - So radiance is independent of direction
 - No participating media (no fog)
 - No transmission (only opaque surfaces)
 - Radiosity is constant across each element
 - Solve for R, G, B separately

Balance of Energy

- Lambertian surfaces (ideal diffuse reflector)
- Divided into n elements
- Variables
 - A_i Area of element i (computable)
 - B_i Radiosity of element i (unknown)
 - E_i Radiant emitted flux density of element i (given)
 - ρ_i Reflectance of element i (given)
 - F_{ji} Form factor from j to i (computable)

$$A_i B_i = A_i E_i + \rho_i \sum_{j=1}^n F_{ji} A_j B_j$$

Form Factors

- **Form factor** F_{ij} : Fraction of light leaving element i arriving at element j
- Depends on
 - Shape of patches i and j
 - Relative orientation of both patches
 - Distance between patches
 - Occlusion by other patches

Form Factor Equation

- Polar angles θ and θ' between normals and ray between x and y
- Visibility function $v(x,y) = 0$ if ray from x to y is occluded, $v(x,y) = 1$ otherwise
- Distance r between x and y

$$A_i F_{ij} = \int_{x \in P_i} \int_{y \in P_j} \frac{\cos \theta \cos \theta'}{\pi r^2} v(x, y) dy dx$$

Reciprocity

- Symmetry of form factor

$$A_i F_{ij} = \int_{x \in P_i} \int_{y \in P_j} \frac{\cos \theta \cos \theta'}{\pi r^2} v(x, y) dy dx = A_j F_{ji}$$

- Divide earlier radiosity equation

$$A_i B_i = A_i E_i + \rho_i \sum_{j=1}^n F_{ji} A_j B_j$$

by A_i

$$\begin{aligned} B_i &= E_i + \rho_i \sum_j (F_{ji} A_j / A_i) B_j \\ &= E_i + \rho_i \sum_j F_{ij} B_j \end{aligned}$$

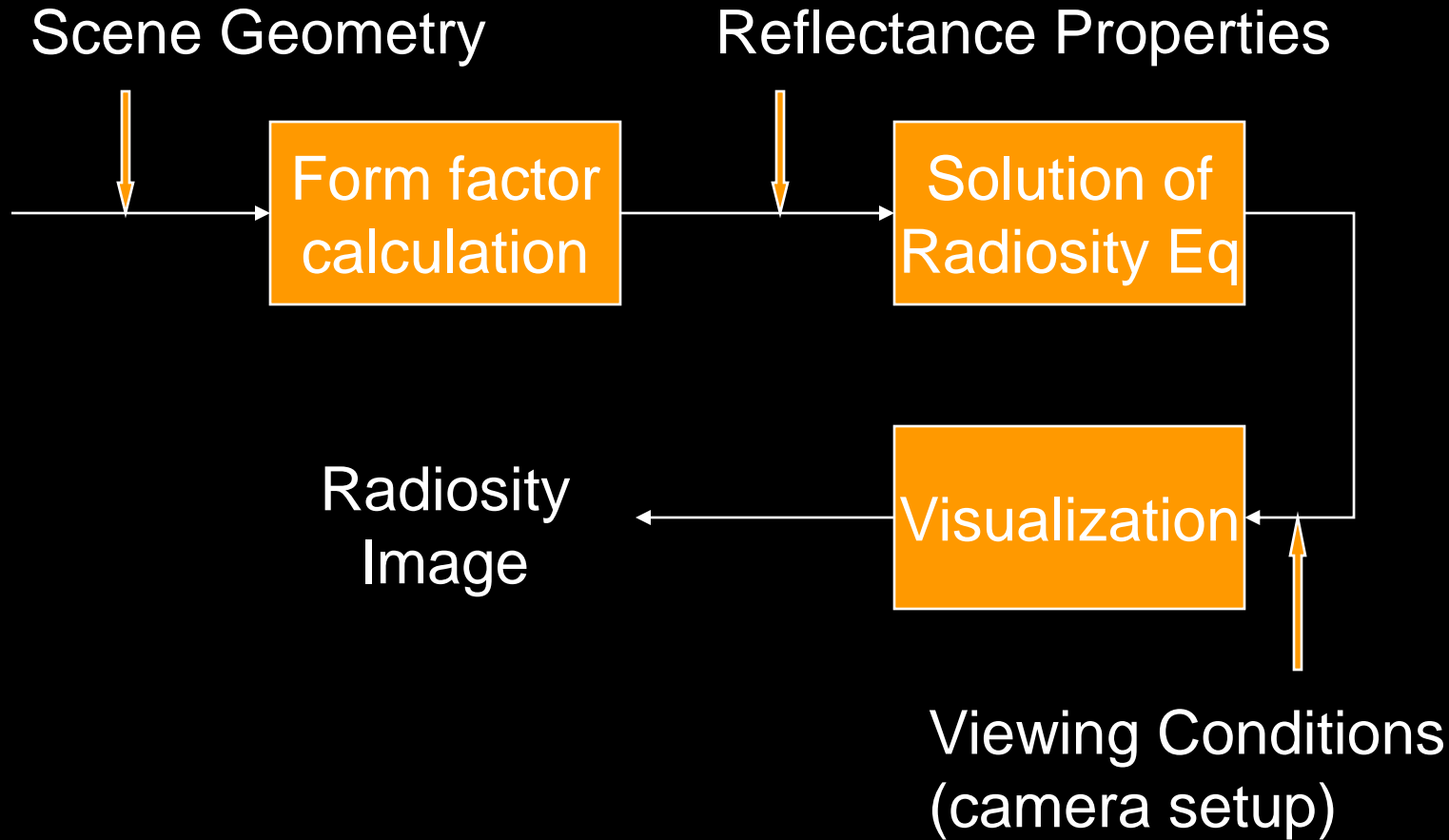
Radiosity as a Linear System

- Restate radiosity equation $B_i - \rho_i \sum_j F_{ij} B_j = E_i$
- In matrix form

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

- Known: reflectances ρ_i , form factors F_{ij} , emissions E_i
- Unknown: Radiosities B_i
- n linear equations in n unknowns

Radiosity "Pipeline"



Visualization

- Radiosity solution is viewer independent
- Can exploit graphics hardware to obtain image
- Convert color on patch to vertex color
- Easy part of radiosity method

Computing Form Factors

- Visibility critical
- Two principal methods
 - Hemicube: exploit z-buffer hardware
 - Ray casting (can be slow)
 - Both exhibit aliasing effects
- For inter-visible elements
 - Many special cases can be solved analytically
 - Avoid full numeric approximation of double integral

Hemicube Algorithm

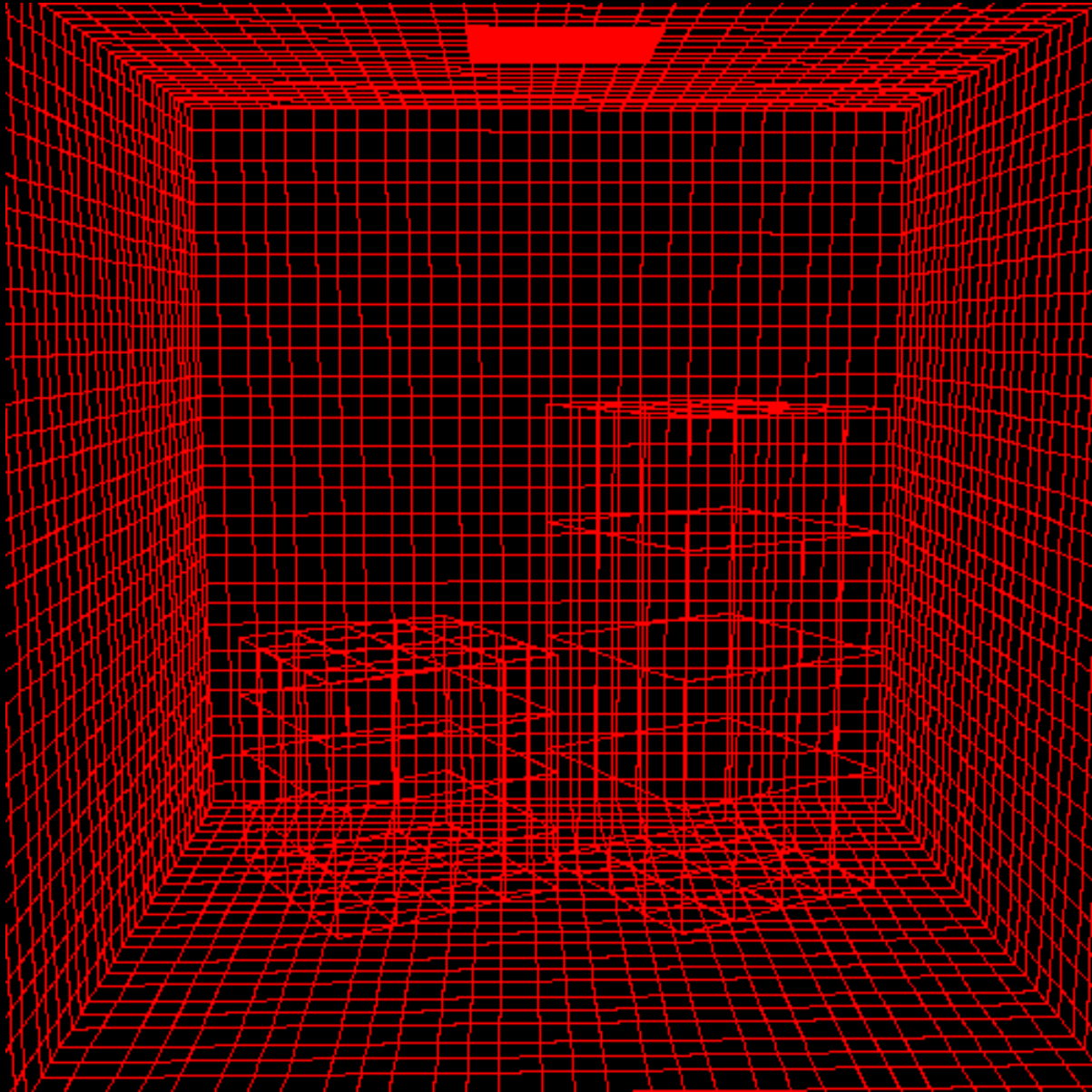
- Render model onto a **hemicube** as seen from the center of a patch
- Store patch identifiers j instead of color
- Use z-buffer to resolve visibility
- Efficiently implementable in hardware

- See Cohen and Greenberg, SIGGRAPH '85

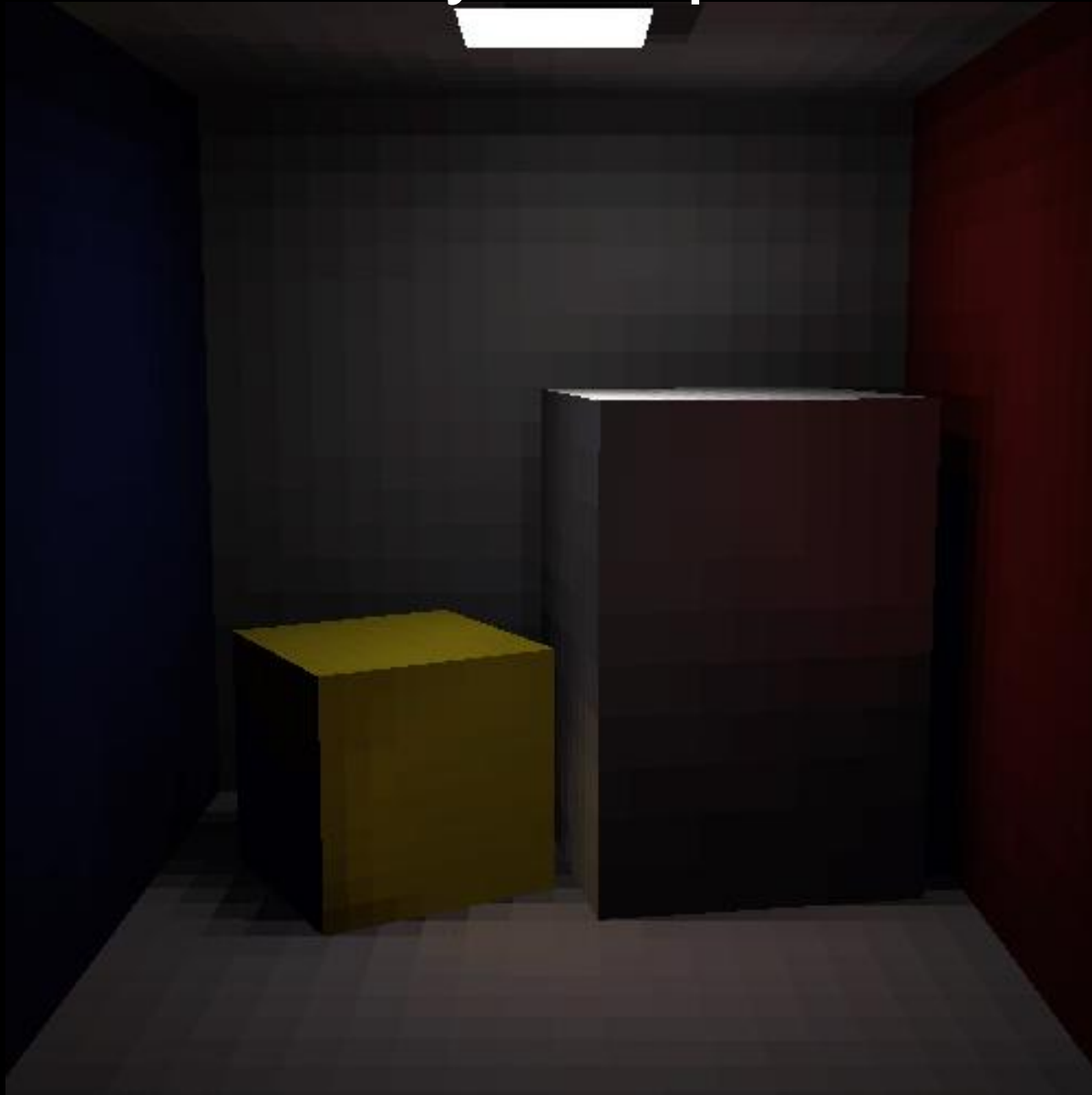
Resolution Issues

- Each patch is a constant color .. how can we obtain smooth, visually pleasing images?

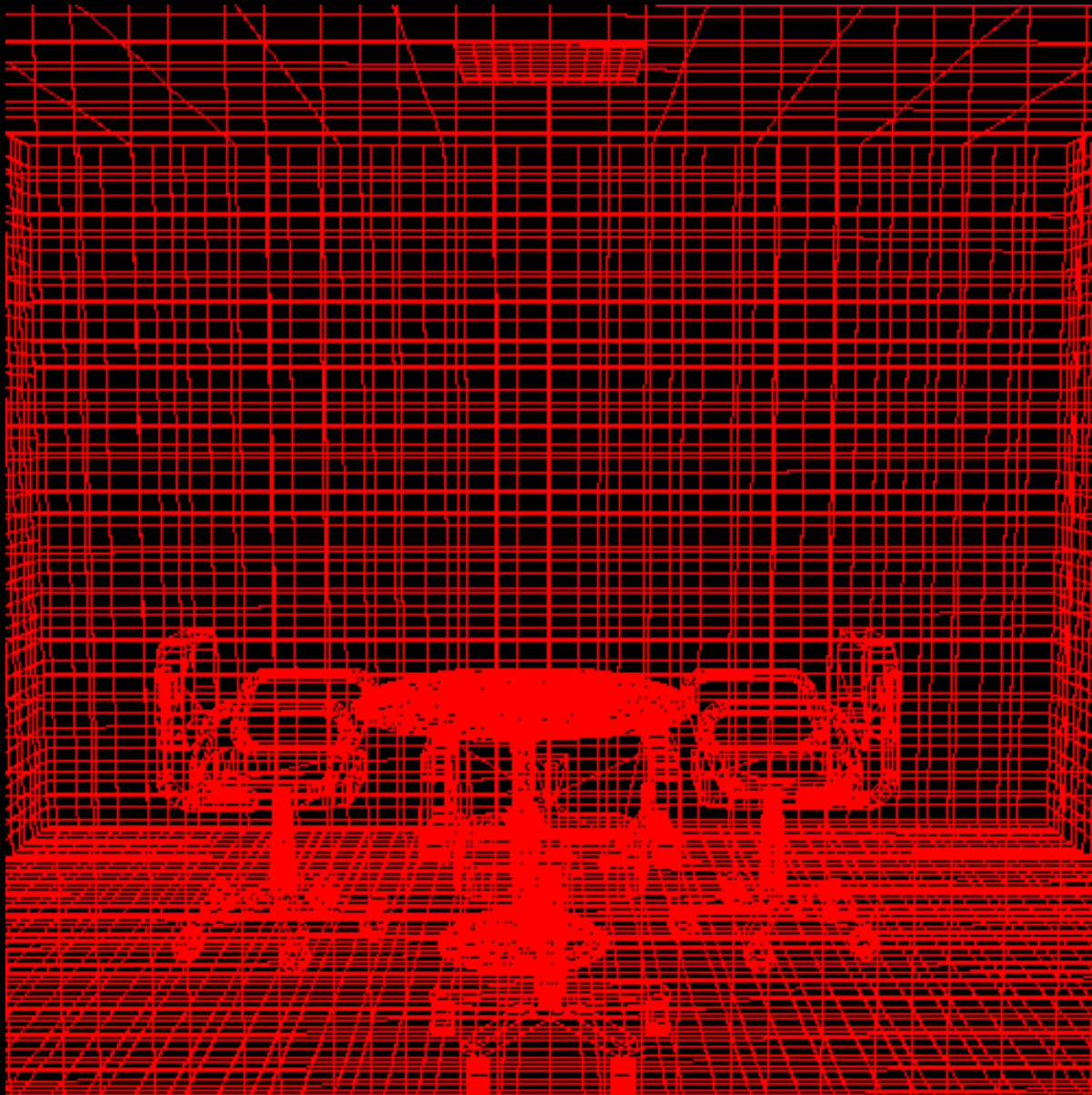
Wireframe



No Intensity Interpolation



Wireframe



Resolution 300



Resolution 1200



Resolution 2500



Resolution 2500, Interpolated



Solving the Radiosity Equation

- Direct form

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

- As matrix equation

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

- Unknown: radiosity B_i
- Known: emission E_i , form factor F_{ij} , reflect. ρ_i

Classical Radiosity Algorithms

- **Matrix Radiosity (Gathering)**
 - Diagonally dominant matrix
 - Use Jacobi's method (iterative solution)
 - Time and space complexity is $O(n^2)$ for n elements
 - Memory cost excessive
- **Progressive Refinement Radiosity (Shooting)**
 - Solve equations incrementally with form factors
 - Time complexity is $O(n \cdot s)$ for s iterations
 - Used more commonly (space complexity $O(n)$)

Matrix Radiosity

- Compute all form factors F_{ij}
- Make initial approximation to radiosity
 - Emitting elements $B_i = E_i$
 - Other elements $B_i = 0$
- Apply equation to get next approximation

$$B'_i = E_i + \rho_i \sum_j F_{ij} B_j$$

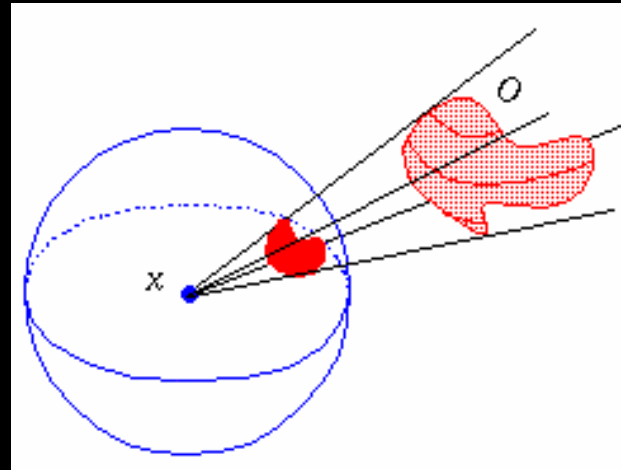
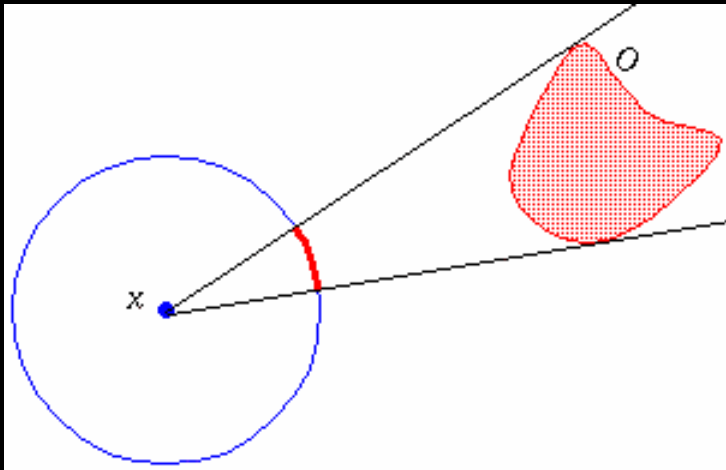
- Iterate with new approximation
- Intuitively
 - Gather incoming light for each element i
 - Base new estimate on previous estimate

Radiosity Summary

- Assumptions
 - Opaque Lambertian surfaces (ideal diffuse)
 - Radiosity constant across each element
- Radiosity computation structure
 - Break scene into patches
 - Compute form factors between patches
 - Lighting independent
 - Solve linear radiosity equation
 - Viewer independent
 - Render using standard hardware

Solid Angle

- 2D **angle** subtended by object O from point x :
 - Length of projection onto unit circle at x
 - Measured in **radians** (0 to 2π)
- 3D **solid angle** subtended by O from point x :
 - Area of of projection onto unit sphere at x
 - Measured in **steradians** (0 to 4π)



Radiant Power and Radiosity

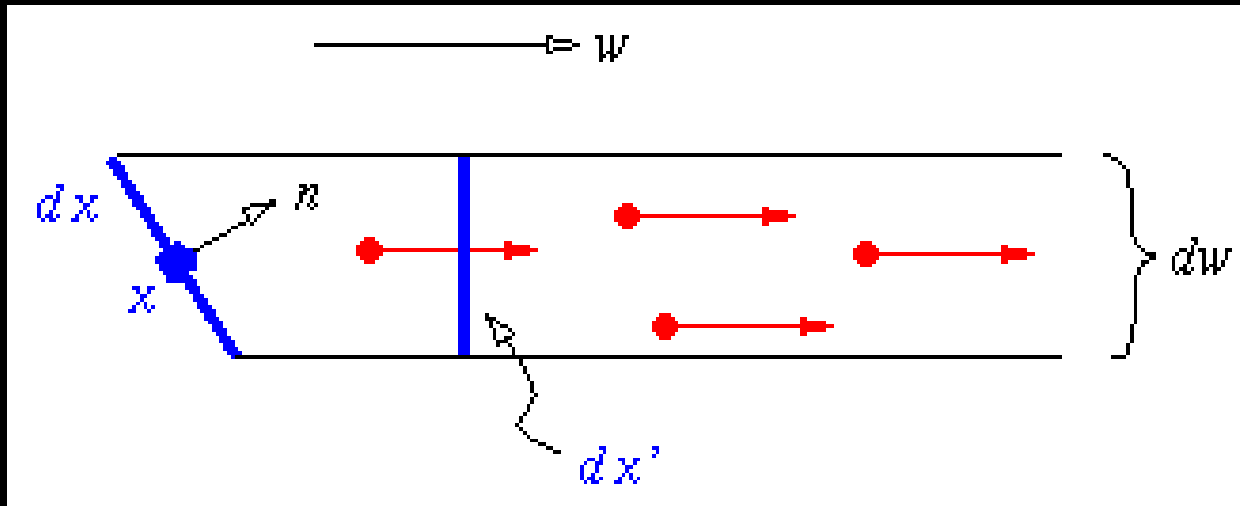
- **Radiant power** P
 - Rate at which light energy is transmitted
 - Dimension: power = energy / time
- **Flux density** Φ
 - Radiant power per unit area of the surface
 - Dimension: power / area
- **Irradiance** E : incident flux density of surface
- **Radiosity** B : exitant flux density of surface
 - Dimension: power / area
- **Flux density at a point** $\Phi(x) = dP/dA$ (or dP/dx)

Power at Point in a Direction

- **Radiant intensity I**
 - Power radiated per unit solid angle by point source
 - Dimension: power / solid angle
- **Radiant intensity in direction ω**
 - $I(\omega) = dP/d\omega$
- **Radiance $L(x, \omega)$**
 - Flux density at point x in direction ω
 - Dimension: power / (area \times solid angle)

Radiance

- Measured across surface in direction ω



J. Stewart '98

- For angle θ between ω and normal n

$$L(x, \omega) = \frac{d^2 P}{d\omega dx'} = \frac{d^2 P}{d\omega \cos \theta dx}$$

Radiosity and Radiance

- Radiosity $B(x) = dP / dx$
- Radiance $L(x, \omega) = d^2P / d\omega \cos\theta dx$
- Let Ω be set of all directions above x

$$B(x) = \int_{\Omega} L(x, \omega) \cos\theta d\omega$$

