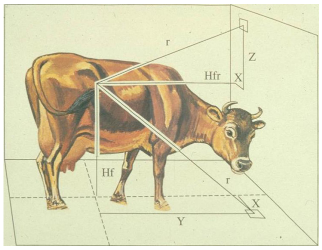
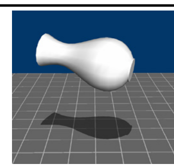
Global Illumination: Radiosity

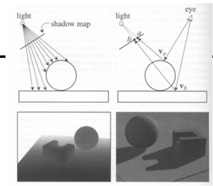


An early application of radiative heat transfer in stables.

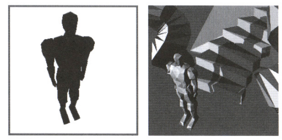
Last Time?



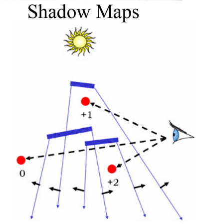
Planar Shadows



Shadow Maps



Projective Texture Shadows
(Texture Mapping)



Shadow Volumes (Stencil Buffer)

Schedule

- Review Session:
  Monday Oct. 25 , 7:30 - 9 pm, Room 1-150
  bring lots of questions!
- Quiz 2: Tuesday October 26th, in class
  80 minutes, closed books, 1 page of notes allowed
- No assignment due next week
- Ray tracing acceleration due Nov 3

Today

- Why Radiosity
  - The Cornell Box
  - Radiosity vs. Ray Tracing
- Global Illumination: The Rendering Equation
- Radiosity Equation/Matrix
- Calculating the Form Factors
- Progressive Radiosity
- Advanced Radiosity

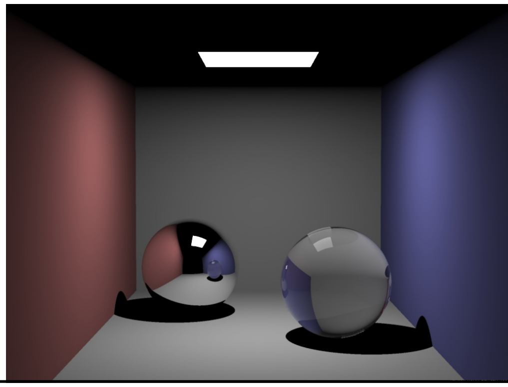
Rendering Recap

- Ray-tracing
  - For each pixel, for each object
- Graphics pipeline, scan conversion
  - For each object, for each pixel
- Local lighting models
  - Diffuse, Phong
- Shadows
  - Ray casting, shadow maps, shadow volumes
- Reflection, refraction

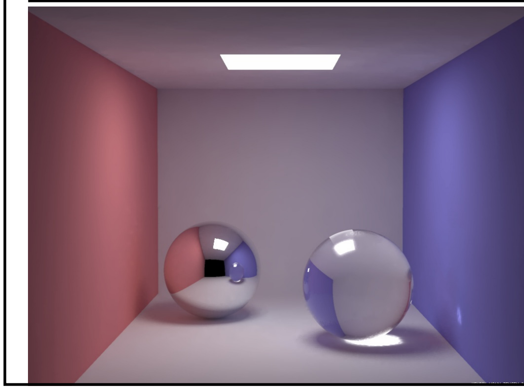
Why global illumination?

- Simulate all light inter-reflections (indirect lighting)
  - e.g. in a room, a lot of the light is indirect: it is reflected by walls.
- How have we dealt with this so far?
  - Ambient term to fake some uniform indirect light

Direct illumination



Global Illumination



Why Radiosity?

- Sculpture by John Ferren
- *Diffuse* panels

photograph:

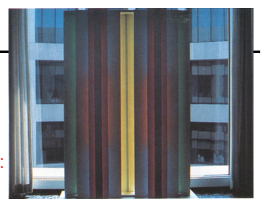
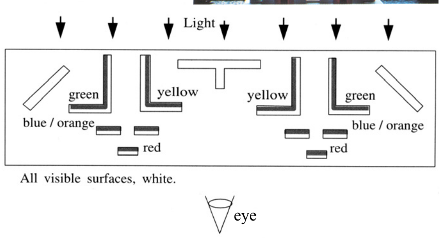
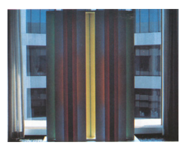


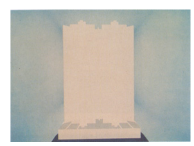
diagram from above:



Radiosity vs. Ray Tracing



Original sculpture by John Ferren lit by daylight from behind.

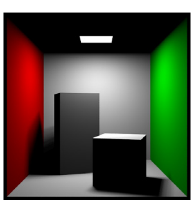


Ray traced image. A standard ray tracer cannot simulate the interreflection of light between diffuse surfaces.

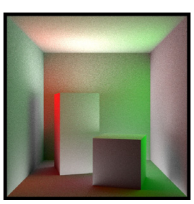
Image rendered with radiosity. note color bleeding effects.

MIT EECS 6.837, Durand and Cutler

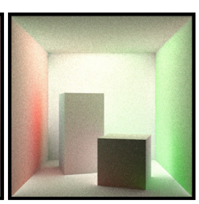
The Cornell Box



direct illumination (0 bounces)



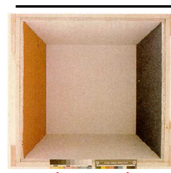
1 bounce



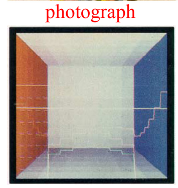
2 bounces

images by Micheal Callahan
http://www.cs.utah.edu/~shirley/classes/cs684_98/students/callahan/bounce/

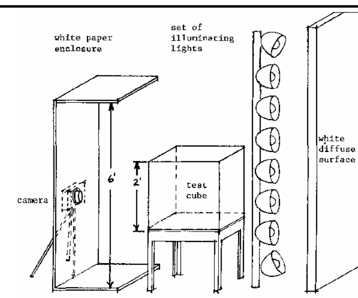
The Cornell Box



photograph



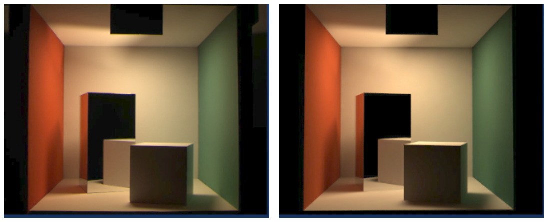
simulation



Goral, Torrance, Greenberg & Battaile
Modeling the Interaction of Light Between Diffuse Surfaces
SIGGRAPH '84

The Cornell Box

- Careful calibration and measurement allows for comparison between physical scene & simulation



photograph simulation

Light Measurement Laboratory
Cornell University, Program for Computer Graphics

Cornell box pun

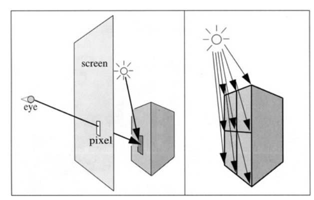
- Cornell university: leading lab in radiosity research in the 80s and 90s
- Joseph Cornell 1903-1973 artist famous for his “boxes”

Two approaches for global illumination

- Radiosity
  - View independent
  - Diffuse only
- Monte-Carlo Ray-tracing
  - Send tons of indirect rays

Radiosity vs. Ray Tracing

- Ray tracing is an *image-space* algorithm
  - If the camera is moved, we have to start over
- Radiosity is computed in *object-space*
  - View independent (just don't move the light)
  - Can pre compute complex lighting to allow interactive walkthroughs



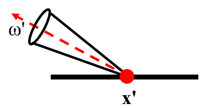
Questions?

Lightscape `http://www.lightscape.com`

Today

- Why Radiosity
  - The Cornell Box
  - Radiosity vs. Ray Tracing
- Global Illumination: The Rendering Equation
- Radiosity Equation/Matrix
- Calculating the Form Factors
- Progressive Radiosity
- Advanced Radiosity

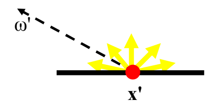
The Rendering Equation



$$L(x',\omega') = E(x',\omega') + \int \rho_{x'}(\omega,\omega') L(x,\omega) G(x,x') V(x,x')\, dA$$

L (x',ω') is the radiance from a point on a surface in a given direction ω'

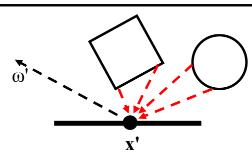
The Rendering Equation



$$L(x',\omega') = E(x',\omega') + \int \rho_{x'}(\omega,\omega') L(x,\omega) G(x,x') V(x,x')\, dA$$

E(x',ω') is the emitted radiance from a point: E is non-zero only if x' is emissive (a light *source*)

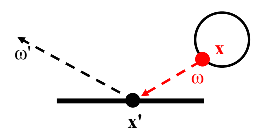
The Rendering Equation



$$L(x',\omega') = E(x',\omega') + \int \rho_{x'}(\omega,\omega') L(x,\omega) G(x,x') V(x,x')\, dA$$

Sum the contribution from all of the other surfaces in the scene

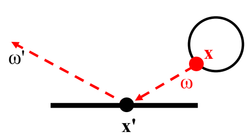
The Rendering Equation



$$L(x',\omega') = E(x',\omega') + \int \rho_{x'}(\omega,\omega') L(x,\omega) G(x,x') V(x,x')\, dA$$

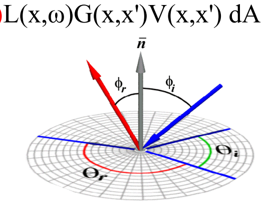
For each x, compute L(x, ω**)**, the radiance at point x in the direction ω (from x to x')

The Rendering Equation

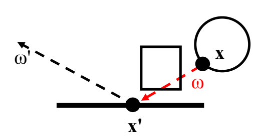


$$L(x',\omega') = E(x',\omega') + \int \rho_{x'}(\omega,\omega') L(x,\omega) G(x,x') V(x,x')\, dA$$

scale the contribution by $\rho_{x'}(\omega,\omega')$, the reflectivity (BRDF) of the surface at x'



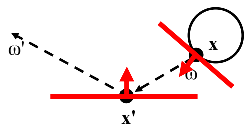
The Rendering Equation



$$L(x',\omega') = E(x',\omega') + \int \rho_{x'}(\omega,\omega') L(x,\omega) G(x,x') V(x,x')\, dA$$

For each x, compute V(x,x'), the visibility between x and x': 1 when the surfaces are unobstructed along the direction ω, 0 otherwise

The Rendering Equation

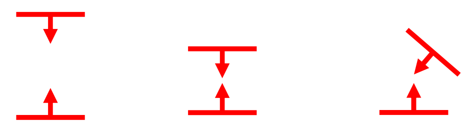


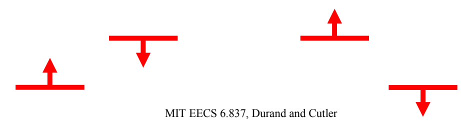
$$L(x',\omega') = E(x',\omega') + \int \rho_{x'}(\omega,\omega')L(x,\omega)G(x,x')V(x,x')\, dA$$

For each x, compute G(x, x'), which describes the on the geometric relationship between the two surfaces at x and x’

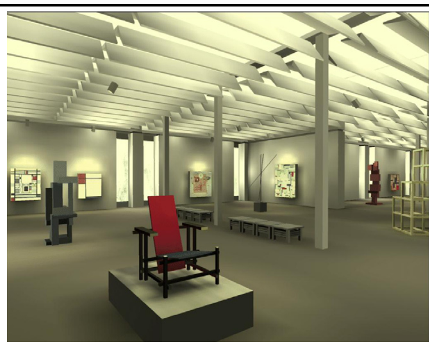
Intuition about G(x,x')?

- Which arrangement of two surfaces will yield the greatest transfer of light energy? Why?





Questions?

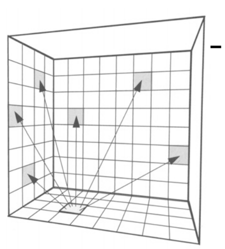


Museum simulation. Program of Computer Graphics, Cornell University.
50,000 patches. Note indirect lighting from ceiling.

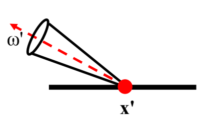
Today

- Why Radiosity
  - The Cornell Box
  - Radiosity vs. Ray Tracing
- Global Illumination: The Rendering Equation
- Radiosity Equation/Matrix
- Calculating the Form Factors
- Progressive Radiosity
- Advanced Radiosity

Radiosity Overview



- Surfaces are assumed to be perfectly Lambertian (diffuse)
  - reflect incident light in all directions with equal intensity
- The scene is divided into a set of small areas, or patches.
- The radiosity, B_i, of patch *i* is the total rate of energy leaving a surface. The radiosity over a patch is constant.
- Units for radiosity: Watts / steradian * meter^2

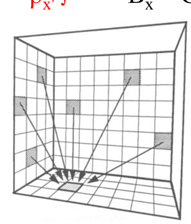


Radiosity Equation

$$L(x',\omega') = E(x',\omega') + \int \rho_{x'}(\omega,\omega')L(x,\omega)G(x,x')V(x,x')\, dA$$

Radiosity assumption:
perfectly diffuse surfaces (not directional)

$$B_{x'} = E_{x'} + \rho_{x'} \int B_x \; G(x,x')V(x,x')$$



Continuous Radiosity Equation

reflectivity

$$B_{x'} = E_{x'} + \rho_{x'} \int G(x,x')\, V(x,x')\, B_x$$

form factor

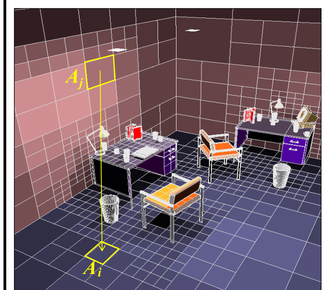
G: geometry term
V: visibility term

No analytical solution,
even for simple configurations

MIT EECS 6.837, Durand and Cutler

Discrete Radiosity Equation

Discretize the scene into *n* patches, over which the radiosity is constant



reflectivity

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

form factor

- discrete representation
- iterative solution
- costly geometric/visibility calculations

MIT EECS 6.837, Durand and Cutler

The Radiosity Matrix

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

n simultaneous equations with *n* unknown B_i values can be written in matrix form:

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

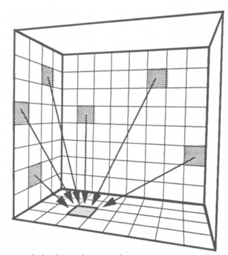
A solution yields a single radiosity value B_i for each patch in the environment, a view-independent solution.

MIT EECS 6.837, Durand and Cutler

Solving the Radiosity Matrix

The radiosity of a single patch *i* is updated for each iteration by *gathering* radiosities from all other patches:

$$\begin{bmatrix} B_1 \\ B_2 \\ \vdots \\ B_i \\ \vdots \\ B_n \end{bmatrix} = \begin{bmatrix} E_1 \\ E_2 \\ \vdots \\ E_i \\ \vdots \\ E_n \end{bmatrix} + \begin{bmatrix} & & & \\ \rho_i F_{i1} & \rho_i F_{i2} & \cdots & \rho_i F_{in} \\ & & & \end{bmatrix} \begin{bmatrix} B_1 \\ B_2 \\ \vdots \\ B_i \\ \vdots \\ B_n \end{bmatrix}$$

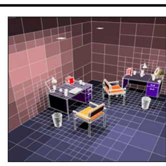


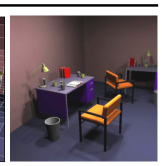
This method is fundamentally a Gauss-Seidel relaxation

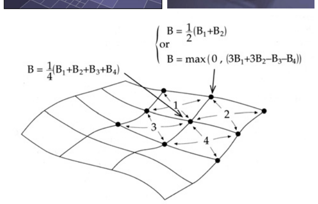
MIT EECS 6.837, Durand and Cutler

Computing Vertex Radiosities

- B_i radiosity values are constant over the extent of a patch.
- How are they mapped to the vertex radiosities (intensities) needed by the renderer?
  - Average the radiosities of patches that contribute to the vertex
  - Vertices on the edge of a surface are assigned values extrapolation







MIT EECS 6.837, Durand and Cutler

Questions?

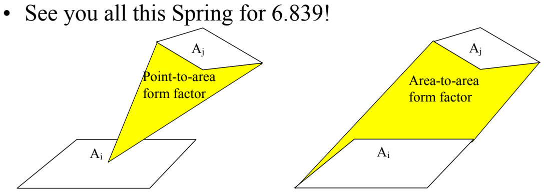
Factory simulation. Program of Computer Graphics, Cornell University.
30,000 patches.

Today

- Why Radiosity
  - The Cornell Box
  - Radiosity vs. Ray Tracing
- Global Illumination: The Rendering Equation
- Radiosity Equation/Matrix
- Calculating the Form Factors
- Progressive Radiosity
- Advanced Radiosity

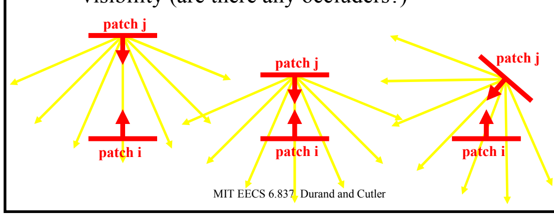
Radiosity Patches are Finite Elements

- We are trying to solve an the rendering equation over the *infinite-dimensional* space of radiosity functions over the scene.
- We project the problem onto a *finite basis* of functions: piecewise constant over patches
- See you all this Spring for 6.839!



Calculating the Form Factor F_{ij}

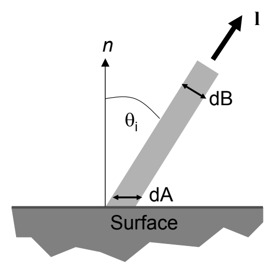
- F_{ij} = fraction of light energy leaving patch j that arrives at patch i
- Takes account of both:
  - geometry (size, orientation & position)
  - visibility (are there any occluders?)



Remember Diffuse Lighting?

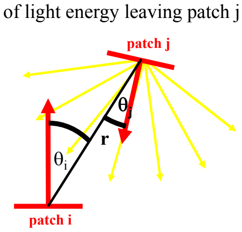
$$L_o = k_d (\mathbf{n} \cdot \mathbf{l}) \frac{L_i}{r^2}$$

$$dA = dB \cos \theta_i$$



Calculating the Form Factor F_{ij}

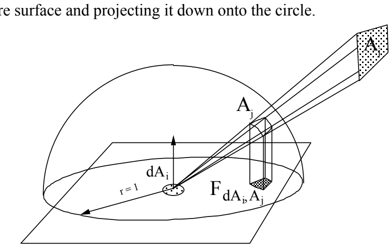
- F_{ij} = fraction of light energy leaving patch j that arrives at patch i



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

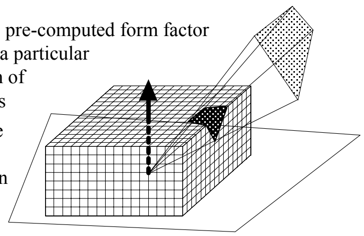
Form Factor Determination

The Nusselt analog: the form factor of a patch is equivalent to the fraction of the the unit circle that is formed by taking the projection of the patch onto the hemisphere surface and projecting it down onto the circle.



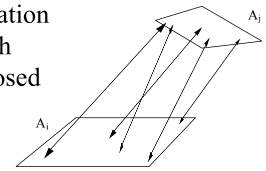
Hemicube Algorithm

- A hemicube is constructed around the center of each patch
- Faces of the hemicube are divided into "pixels"
- Each patch is projected (rasterized) onto the faces of the hemicube
- Each pixel stores its pre-computed form factor The form factor for a particular patch is just the sum of the pixels it overlaps
- Patch occlusions are handled similar to z-buffer rasterization



Form Factor from Ray Casting

- Cast n rays between the two patches
  - n is typically between 4 and 32
  - Compute visibility
  - Integrate the point-to-point form factor
- Permits the computation of the patch-to-patch form factor, as opposed to point-to-patch



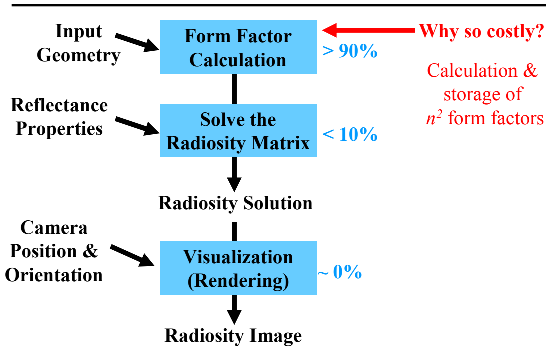
Questions?

Lightscape `http://www.lightscape.com`

Today

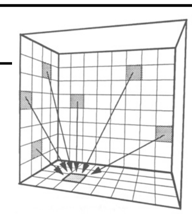
- Why Radiosity
  - The Cornell Box
  - Radiosity vs. Ray Tracing
- Global Illumination: The Rendering Equation
- Radiosity Equation/Matrix
- Calculating the Form Factors
- Progressive Radiosity
- Advanced Radiosity

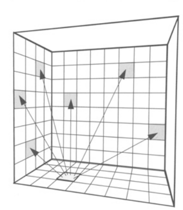
Stages in a Radiosity Solution



Progressive Refinement

- Goal: Provide frequent and timely updates to the user during computation
- Key Idea: Update the entire image at every iteration, rather than a single patch
- How? Instead of summing the light received by one patch, distribute the radiance of the patch with the most *undistributed radiance*.

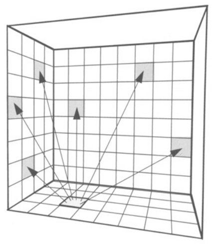




Reordering the Solution for PR

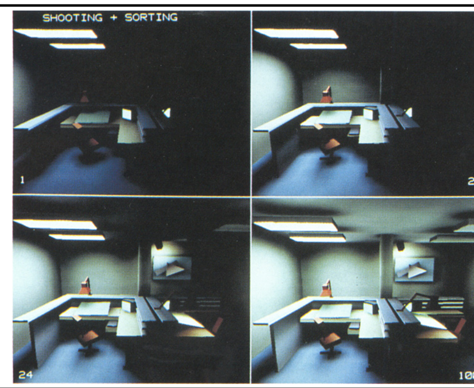
Shooting: the radiosity of all patches is updated for each iteration:

$$\begin{bmatrix} B_1 \\ B_2 \\ \vdots \\ \\ \vdots \\ B_n \end{bmatrix} = \begin{bmatrix} B_1 \\ B_2 \\ \vdots \\ \\ \vdots \\ B_n \end{bmatrix} + \begin{bmatrix} \cdots & \rho_1 F_{1i} & \cdots \\ \cdots & \rho_2 F_{2i} & \\ & & \\ & & \\ & & \\ \cdots & \rho_n F_{ni} & \cdots \end{bmatrix} \begin{bmatrix} \\ \\ \vdots \\ B_i \\ \vdots \\ \\ \end{bmatrix}$$

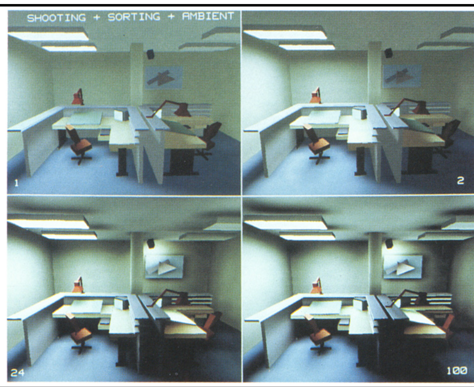


This method is fundamentally a Southwell relaxation

Progressive Refinement w/out Ambient Term



Progressive Refinement with Ambient Term



Questions?

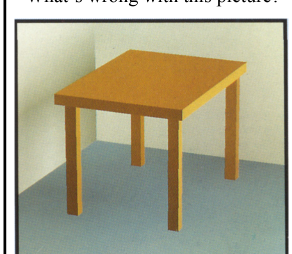
Lightscape `http://www.lightscape.com`

Today

- Why Radiosity
- Global Illumination: The Rendering Equation
- Radiosity Equation/Matrix
- Calculating the Form Factors
- Progressive Radiosity
- Advanced Radiosity
  - Adaptive Subdivision
  - Discontinuity Meshing
  - Hierarchical Radiosity
  - Other Basis Functions

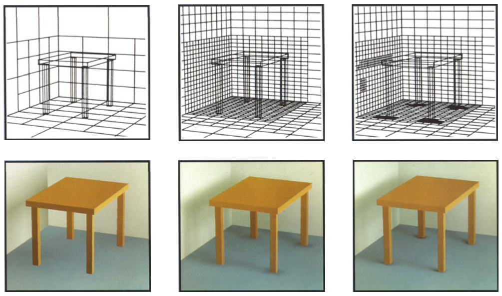
Increasing the Accuracy of the Solution

What's wrong with this picture?



- The quality of the image is a function of the size of the patches.
- The patches should be *adaptively subdivided* near shadow boundaries, and other areas with a high radiosity gradient.
- Compute a solution on a uniform initial mesh, then refine the mesh in areas that exceed some error tolerance.

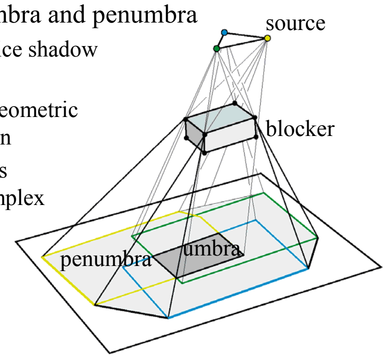
Adaptive Subdivision of Patches



Coarse patch solution (145 patches) | Improved solution (1021 subpatches) | Adaptive subdivision (1306 subpatches)

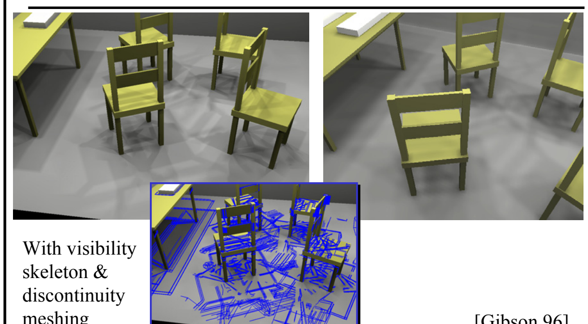
Discontinuity Meshing

- Limits of umbra and penumbra
  - Captures nice shadow boundaries
  - Complex geometric computation
  - The mesh is getting complex



Discontinuity Meshing

Discontinuity Meshing Comparison



With visibility skeleton & discontinuity meshing

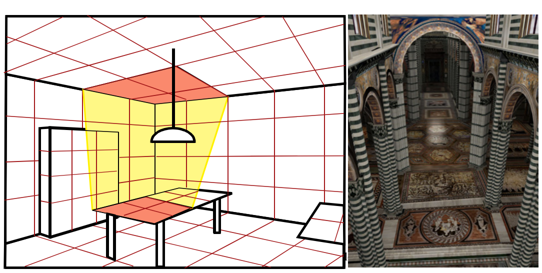
10 minutes 23 seconds

[Gibson 96]

1 hour 57 minutes

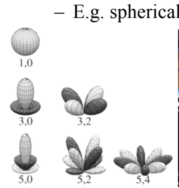
Hierarchical Approach

- Group elements when the light exchange is not important
  - Breaks the quadratic complexity
  - Control non trivial, memory cost



Other Basis Functions

- Higher order (non constant basis)
  - Better representation of smooth variations
  - Problem: radiosity is discontinuous (shadow boundary)
- Directional basis
  - For non-diffuse finite elements
  - E.g. spherical harmonics



Questions?

Lightscape `http://www.lightscape.com`

Radiosity today

- Used in architectural simulation (Lightscape software)
- Used for game lighting preprocessing (light maps)

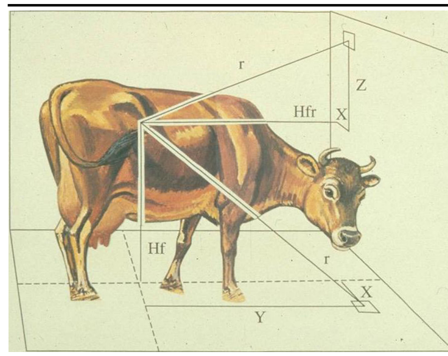
- Not as hot a research topic
  - Monte Carlo Ray-tracing is hotter (more general)
  - But "pre-computed radiance transfer" is very close: idea of projecting onto simpler basis functions (used e.g. in Max Payne 2)

Practical problems with radiosity

- Meshing (memory, robustness)
- Form factors (computation)
- Diffuse limitation (extension to specular takes too much memory)

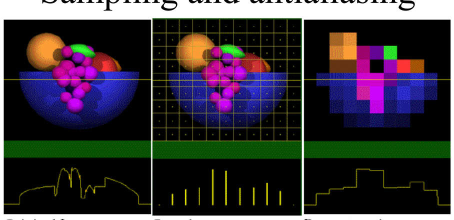
- Fast extensions (hierarchical) can be hard to control

Cow-cow form factor



Next Time:

Quiz!
And then, Thursday:
Sampling and antialiasing



Original Image Samples Reconstruction