CS770/870 Spring 2017
Radiosity

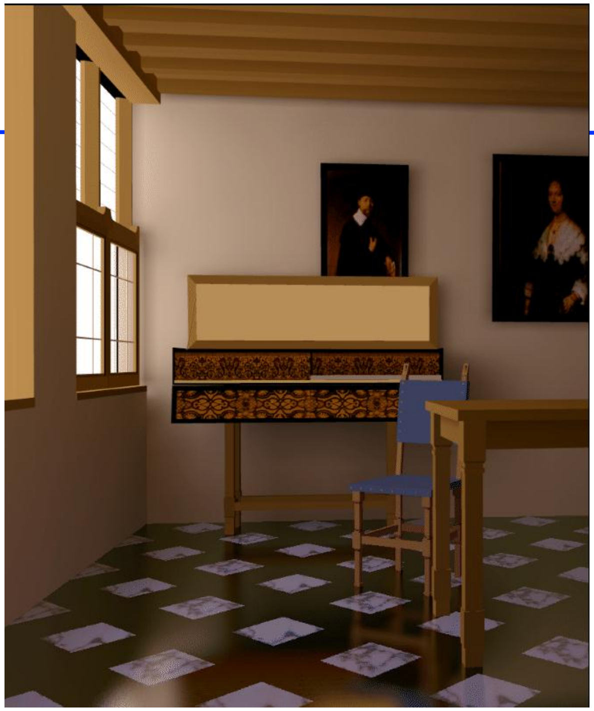
Greenberg, SIGGRAPH '86 Tutorial
Spencer, SIGGRAPH '93 Slide Set, siggraph.org/education/materials/HyperGraph/radiosity/radiosity.htm
Watt, *3D Computer Graphics -- Third Edition*, Addison-Wesley, 2000.

Preview

- Indirect light models
- Brief radiosity overview
- Radiosity details
  - bidirectional reflectance
  - radiosity equation
  - radiosity approximation
- Implementation details
  - hemicube

Models for Indirect Light

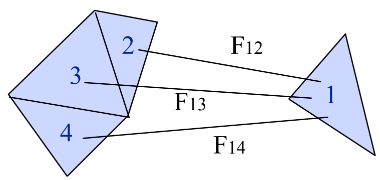
- Interreflection of objects in scene
  - light reflected from each object can affect brightness of other objects
- Ray tracing models indirect *specular* reflection
  - following rays along direction of principal reflection models interreflection of shiny surfaces
- Radiosity models indirect *diffuse* reflection
  - originally developed to model heat transfer
  - first applied to computer graphics by Don Greenberg and students at Cornell in early 80's



Cornell Program in Computer Graphics

Radiosity Algorithm — Setup

- Light sources and objects treated alike
- Partition all objects into small <u>*patches*</u>
- Find interreflection between each patch pair — called *form factors* — includes visibility
- F_{12} = % light leaving patch 1 that reaches patch 2
- Patch 2 reflects R_2 % of light that lands on it, so
- Patch 2 reflects $R_2 * F_{12}$ light shot from Patch 1



Bidirectional Reflectance

- Relationship of light energy arriving at a point, P_i, to the light leaving that point

  $\rho = I_r / E_i$

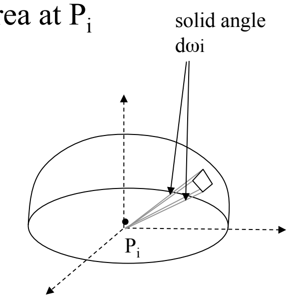
  where I_r is reflected intensity per unit area

  and E_i is incoming energy per unit area at P_i
- What is the incoming energy, E_i ?

  $E_i = f(I_i, d\omega_i)$, some function

  where I_i is the incoming light *intensity*

  and $d\omega_i$ is solid angle through which intensity arrives
  - What is the f? What is I_i?



What is Radiosity?

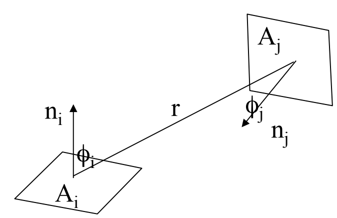
- Radiosity (B): energy per unit area leaving a surface per unit time; it is the sum of:
  - energy *emitted* at that point
  - energy *reflected* at that point
- Radiosity of a <u>patch</u> i whose area is dA_i:

$$B_i dA_i = E_i dA_i + R_i \int_j B_j F_{ji} dA_j$$

  - E_i is energy emitted by patch I per unit area
  - R_i is the % of incident energy that is reflected (per wavelength)
  - F_{ji} is the *form factor* that defines the relationship between patch *j* and patch *i*.

Form Factor

- What is the *form factor* from patch *i* to patch *j*?
  - Assume all surfaces are perfect Lambertian diffuse reflectors
    - I.e., all incident energy is reflected *equally* in all directions.
  - F_{ij} is the percent of energy leaving patch *i* that actually hits *j*
    - *occlusion*: if another patch is between *i* and *j*, $F_{ij} == 0$; vis_{ij}
    - also depends on the relative orientation of the 2 patches:



$$f_{ij}(dA_i, dA_j) = \frac{\cos\phi_i \cos\phi_j}{\pi r^2} dA_i dA_j$$

So, $f_{ij}(dA_i, A_j) = dA_i \frac{\cos\phi_i}{\pi r^2} \int_{A_j} \cos\phi_j dA_j$

and $f_{ij}(A_i, A_j) = \int_{A_i}\int_{A_j} \frac{\cos\phi_i \cos\phi_j}{\pi r^2} dA_j dA_i$

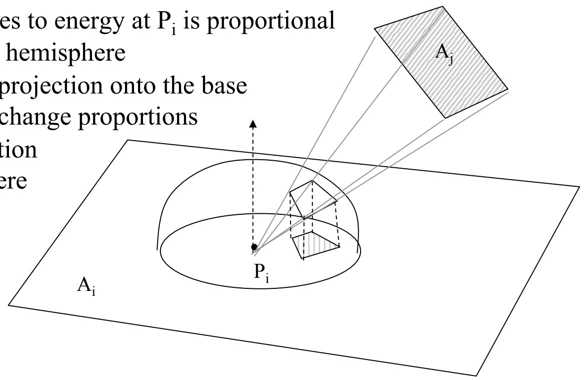
To make unitless, divide by A_i; also add occlusion test

$$F_{ij}(A_i, A_j) = \frac{vis_{ij}}{A_i} \int_{A_i}\int_{A_j} \frac{\cos\phi_i \cos\phi_j}{\pi r^2} dA_i dA_j$$

What does it mean?

Geometric Interpretation of Form Factor Integral

- Light is reflected by patch *i* equally in all directions
  - represent reflection by hemisphere with r = 1 around patch center
  - all energy arriving at P_i also comes through the hemisphere
  - project A_j onto hemisphere
  - contribution of other patches to energy at P_i is proportional to their projection onto the hemisphere
  - projecting the hemisphere projection onto the base of the hemisphere doesn't change proportions
  - form factor equals the fraction of the base of the hemisphere covered by the projection.

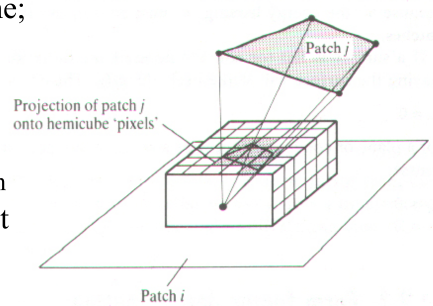


Calculating Form Factors

- Greenberg's group at Cornell borrowed radiosity from the heat transfer field
  - image synthesis, however, needs much higher patch resolution than heat transfer modeling
  - their principal initial contribution was a tractable algorithm for computing form factors
  - the *hemicube* is a rectangular approximation to the hemisphere that is placed over the center of each patch

Hemicube

- Each face of the hemicube is projected onto the hemicube base
  - Compute *delta form factors* for each hemicube "pixel" versus the hemicube base; analytic approach yields a closed form!
  - Given hemicube size and resolution, delta form factors are the same for *all* hemicubes for all time; they are pre-computed and saved
  - Project each patch onto hemicube
    - form factor for the patch is the sum of the delta form factors of all hemicube pixels covered by the patch
  - A major computational component



Radiosity Equation Revisited

$$B_i dA_i = E_i dA_i + R_i \int_j B_j F_{ji} dA_j$$

But, there is a reciprocity between F_{ji} and F_{ij} : $F_{ij} A_i = F_{ji} A_j$

So, $B_i dA_i = E_i dA_i + R_i \int_j B_j F_{ij} dA_i$

Dividing by dA_i yields: $B_i = E_i + R_i \int_j B_j F_{ij}$

This can be approximated by: $B_i = E_i + R_i \sum_j B_j F_{ij}$

The E_i are known, so rewrite as:

$$B_i - R_i \sum_j B_j F_{ij} = E_i$$

Radiosity Solution

- There are n radiosity equations, one for patch
  - Leads to n simultaneous linear equations to be solved

$$\begin{bmatrix} 1-R_1F_{11} & -R_1F_{12} & \cdots & -R_1F_{1n} \\ -R_2F_{21} & 1-R_2F_{22} & \cdots & -R_2F_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ -R_nF_{n1} & -R_nF_{12} & \cdots & 1-R_nF_{nn} \end{bmatrix} \begin{bmatrix} B_1 \\ B_2 \\ \vdots \\ B_n \end{bmatrix} = \begin{bmatrix} E_1 \\ E_2 \\ \vdots \\ E_n \end{bmatrix}$$

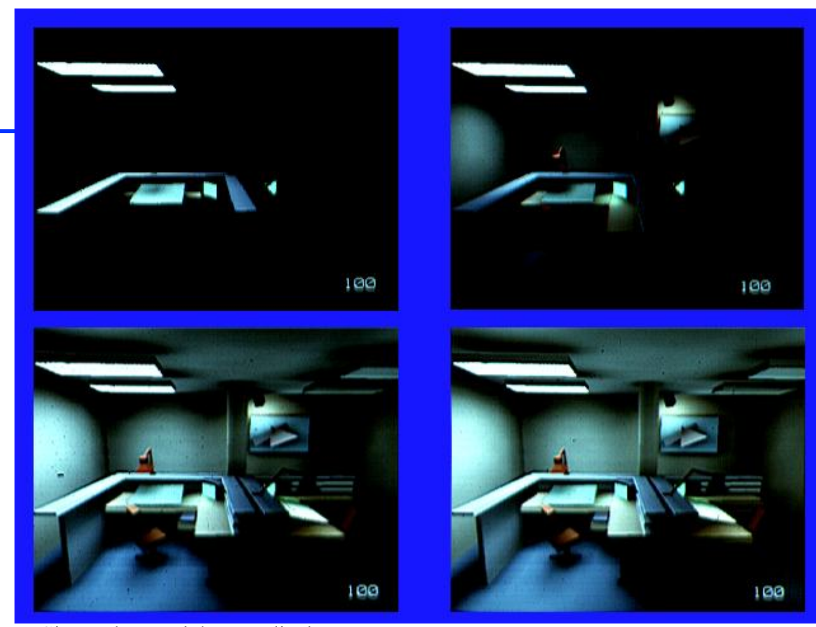
  - E_i are non-zero only for light patches, R_i are known
  - Can solve with something like *Gauss-Seidel* method
    - guaranteed to converge rapidly for this kind of data
    - in effect, it looks at each patch, *gathers* current contributions from all other patches to update this patch and iterates
    - can think of it as *breadth-first* approach

Progressive Refinement: Shooting

- In 1988, Cornell group developed a progressive refinement solution that rendered while computing
  - the *gather* approach of *Gauss-Seidel* was much too slow
  - they changed the focus to a *shooting* one
- Choose a patch
  - determine its contribution to all other patches
  - update the image
  - iterate until "done"
- Key difference is that can choose patches adaptively:
  - initially want to choose light patches
  - in general, choose a patch that will have most effect

Adaptive Shooting

- Associate *unshot* light with each patch — initially set to light *emitted* by the patch
  - for each patch with unshot light, *shoot* that light to all other patches (using form factor)
  - each patch *gathers* light shot to it — what it gathers at one stage is added to its *unshot* light for next stage
- At each step, pick the patch with the most unshot light
- Repeat until all patches have very little unshot light
- Add a global ambient light term
  - start with an estimate based on total light in scene
  - reduce the ambient term at each iteration based on unshot light



Siggraph Tutorial on Radiosity

Radiosity Algorithm

```
Partition all objects into patches
Compute form factors between patches
globalAmbient = estimateAmbient()
maxPatch = findMaxUnshotPatch()
while ( maxPatch.unshot > minimum )
  foreach patch (except this one)
     update B with this unshot light
  reduce globalAmbient
  rerender image
  maxPatch = findMaxUnshotPatch()
```


Radiosity Algorithm Updates

- The entire algorithm does not always need to be executed. What happens if something change:
  - Viewpoint
    - None of radiosity calculations change; only need to redraw!
  - Light intensity/colors
    - No form factors change, so only need to re-shoot the light
  - Geometry changes
    - requires form factor re-computation; this is the slowest part

Radiosity Characteristics

- Strengths
  - Highly realistic diffuse reflection, including soft shadows, and area light sources
- Weakness: can't easily address specular inter-reflection
- Artifact issues
  - Hemicube-based artifacts
    - aliasing can occur because of the regular subdivision nature of the hemicube
    - aliasing can occur where a light patch is close to other patches
    - all issues can be ameliorated with higher resolution, and higher cost
  - Interpolation issues
  - Meshing issues

Review

- Radiosity
  - bidirectional reflectance
  - radiosity equation
  - radiosity approximation
- Implementation details
  - hemicube
  - gather approach
  - shooting approach