University of BRISTOL

COMS 30115

Classic Radiosity

Carl Henrik Ek - carlhenrik.ek@bristol.ac.uk

March 18th, 2018

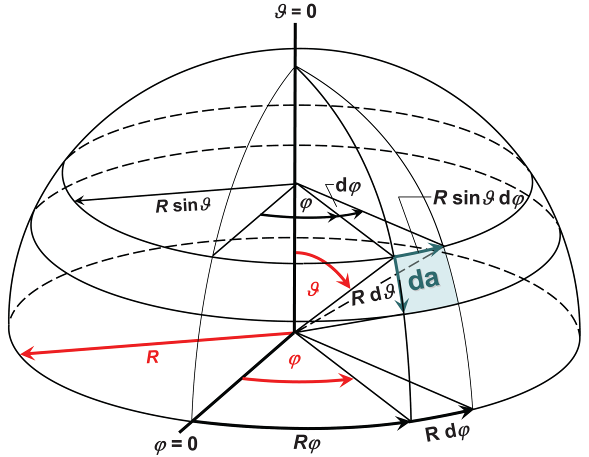
http://www.carlhenrik.com

Introduction

Last Time

- Hemisphere
- BRDFs
- Rendering Equation

Hemisphere



BRDFs

- The BRDF tells us how to represent reflected radiance in terms of incoming radiance

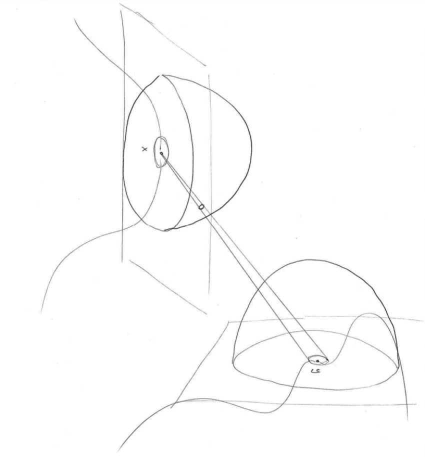
$$L_r(x \to \Theta) = \int_{\Omega_x} f_r(x, \Psi \to \Theta) L(x \leftarrow \Psi) cos(\mathbf{n}_x, \Psi) d\omega_\Psi$$

- Putting these together gives the rendering equation,

$$\begin{aligned} L(x \to \Theta) &= L_e(x \to \Theta) \\ &\quad + \int_{\Omega_x} f_r(x, \Psi \to \Theta) L(x \leftarrow \Psi) cos(\mathbf{n}_x, \Psi) d\omega_\Psi \end{aligned}$$

- We want to solve for the radiance $L(\forall x \to \forall \theta)$ for the whole scene, why is this complicated?

Rendering Equation

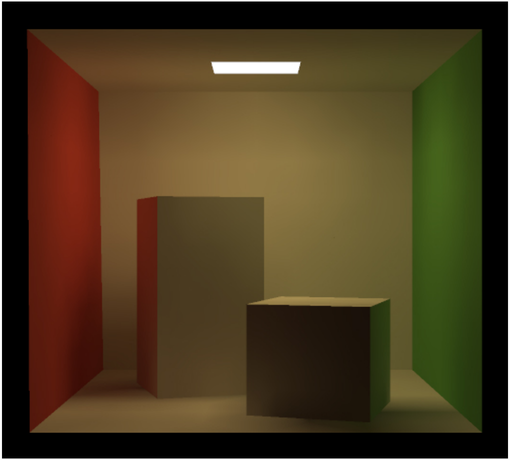


$$L(x \rightarrow \Theta) = L_e(x \rightarrow \Theta) + \int_{\Omega_x} f_r(x, \Psi \rightarrow \Theta) L(x \leftarrow \Psi) cos(\mathbf{n}_x, \Psi) d\omega_\Psi$$

Today

- Area formulation of the rendering equation
- Classic Radiosity

Cornell Box



Material

- Comparing two Global Illumination Models
- Radiosity
- Equation Compendium

Transport Problem

Transport Problem

- Our eyes sensitive to radiance
- Want to compute radiance from all “surfaces” & solid angles
- Flux is emitted from light sources
- The rendering equation tells us how this flux is distributed in the scene

Transport Problem

$$L(x \to \Theta) = L_e(x \to \Theta) \\ + \int_{\Omega_x} f_r(x, \Psi \to \Theta) L(x \leftarrow \Psi) cos(\mathbf{n}_x, \Psi) d\omega_\Psi$$

Transport Problem

$$
\begin{aligned}
L(x \to \Theta) &= L_e(x \to \Theta) \\
&\quad + \int_{\Omega_x} f_r(x, \Psi \to \Theta) L(x \leftarrow \Psi) cos(\mathbf{n}_x, \Psi) d\omega_\Psi \\
&= L_e(x \to \Theta) + \mathcal{T}(L(x \to \Theta))
\end{aligned}
$$

Transport Problem

$$
\begin{aligned}
L(x \to \Theta) &= L_e(x \to \Theta) \\
&\quad + \int_{\Omega_x} f_r(x, \Psi \to \Theta) L(x \leftarrow \Psi) cos(\mathbf{n}_x, \Psi) d\omega_\Psi \\
&= L_e(x \to \Theta) + \mathcal{T}(L(x \to \Theta)) \\
\mathcal{T}(L(x \to \Theta)) &= \int_{\Omega_x} f_r(x, \Psi \to \Theta) L(x \leftarrow \Psi) cos(\mathbf{n}_x, \Psi) d\omega_\Psi
\end{aligned}
$$

Transport Problem

$$\mathbf{x}^{\mathrm{T}}\mathbf{a} = \sum_{i=1}^{D} x_i \cdot a_i, \quad \langle f, g \rangle = \int f(x)g(x)\mathrm{d}x$$

- Rendering Equation formulates recursive transport
- Transport is a linear operator in radiance

Transport Problem

$$\langle F^{\rightarrow}, G^{\leftarrow} \rangle = \int_A \int_\Omega F(x \rightarrow \Theta) G(x \leftarrow \Theta) cos(\mathbf{n}_x, \Theta) \mathrm{d}\omega_\Theta \mathrm{d}A$$

Inner product formulation

- Transport problem can be written as the inner-product of two functions

Transport Problem

$$L(x \to \Theta) = L = L_e + \langle T, L \rangle$$

Transport Problem

$$\begin{aligned} L(x \to \Theta) = L &= L_e + \langle T, L \rangle \\ &= L_e + \langle T, L_e \rangle + \langle T, TL \rangle \end{aligned}$$

Transport Problem

$$
\begin{aligned}
L(x \to \Theta) = L &= L_e + \langle T, L \rangle \\
&= L_e + \langle T, L_e \rangle + \langle T, TL \rangle \\
&= L_e + \langle T, L_e \rangle + \langle T, TL_e \rangle + \langle T, TTL_e \rangle + \dots
\end{aligned}
$$

Transport Problem

$$
\begin{aligned}
L(x \to \Theta) = L &= L_e + \langle T, L \rangle \\
&= L_e + \langle T, L_e \rangle + \langle T, TL \rangle \\
&= L_e + \langle T, L_e \rangle + \langle T, TL_e \rangle + \langle T, TTL_e \rangle + \dots
\end{aligned}
$$

- Adjoint operator:

$$\forall F, G \in V : \langle \mathcal{O}_1 F, G \rangle = \langle F, \mathcal{O}_2 G \rangle$$

Transport Problem

$$
\begin{aligned}
L(x \to \Theta) = L &= L_e + \langle T, L \rangle \\
&= L_e + \langle T, L_e \rangle + \langle T, TL \rangle \\
&= L_e + \langle T, L_e \rangle + \langle T, TL_e \rangle + \langle T, TTL_e \rangle + \dots
\end{aligned}
$$

- Adjoint operator:

$$\forall F, G \in V : \langle \mathcal{O}_1 F, G \rangle = \langle F, \mathcal{O}_2 G \rangle$$

- *Think raytracer!*
- Computing light transport back to light source or from light source are adjoint operators
- Remember dual formulation of raytracing

Transport Problem

How to proceed

1. Input: geometry and emitted light

Transport Problem

How to proceed

1. Input: geometry and emitted light
2. Output: radiance for position & solid angle

Transport Problem

How to proceed

1. Input: geometry and emitted light
2. Output: radiance for position & solid angle
3. What is the set of points & solid angles?

Transport Problem

How to proceed

1. Input: geometry and emitted light
2. Output: radiance for position & solid angle
3. What is the set of points & solid angles?
    - what is the perceptual importance of each part of the geometry?

Transport Problem

How to proceed

1. Input: geometry and emitted light
2. Output: radiance for position & solid angle
3. What is the set of points & solid angles?
    - what is the perceptual importance of each part of the geometry?
4. How do we perform series expansion?

Transport Problem

How to proceed

1. Input: geometry and emitted light
2. Output: radiance for position & solid angle
3. What is the set of points & solid angles?
    - what is the perceptual importance of each part of the geometry?
4. How do we perform series expansion?
    - How does normal raytracing do this?

Transport Problem

How to proceed

1. Input: geometry and emitted light
2. Output: radiance for position & solid angle
3. What is the set of points & solid angles?
    - what is the perceptual importance of each part of the geometry?
4. How do we perform series expansion?
    - How does normal raytracing do this?
    - Perceptual importance?

Transport Problem

How to proceed

1. Input: geometry and emitted light
2. Output: radiance for position & solid angle
3. What is the set of points & solid angles?
    - what is the perceptual importance of each part of the geometry?
4. How do we perform series expansion?
    - How does normal raytracing do this?
    - Perceptual importance?
5. Different algorithms approach this in different way

Transport Problem

How to proceed

1. Input: geometry and emitted light
2. Output: radiance for position & solid angle
3. What is the set of points & solid angles?
   - what is the perceptual importance of each part of the geometry?
4. How do we perform series expansion?
   - How does normal raytracing do this?
   - Perceptual importance?
5. Different algorithms approach this in different way
6. We know that energy is balanced in the scene

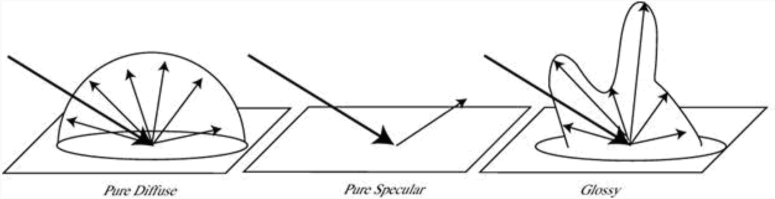
Radiosity

Radiosity

- Initially developed for solving heat transfer
- Formulation for Computer Graphics[1]
- Game engine Geomerics[2]

[1] Goral, C. M., Torrance, K. E., Greenberg, D. P., & Battaile, B. (1984).
[2] http://www.geomerics.com/

Radiosity



- Diffuse relfections means that reflection is the same in all directions
- This means radiance is the same in each direction
- This simplifies the problem rendering problem hugely
- Classic Radiosity assumes all surfaces are perfectly diffuse

Radiosity

Radiosity the exitant radiant power on a surface per unit surface area

- B(x) - $[Watt/m^2]$

Radiance the radiant power per unit projected area per unit solid angle

- L(x) - $[Watt/steradian \cdot m^2]$

If we are assuming each surface to be diffuse we do not need the angle

$$B_i = \frac{1}{A_i} \int_{S_i} \int_{\Omega_x} L(x \rightarrow \Theta) cos(\mathbf{n}_x, \Theta) \mathrm{d}\omega_\Theta \mathrm{d}A_x$$

Radiosity

$$L(x \to \Theta) = L_e(x \to \Theta) \\ + \int_{\Omega_x} f_r(x, \Psi \to \Theta) L(x \leftarrow \Psi) cos(\mathbf{n}_x, \Psi) d\omega_\Psi$$

- Angles does not matter if we have perfectly diffuse surfaces
- Write the rendering equation in terms of surfaces instead

Radiosity

$$
\begin{aligned}
L(x \to \Theta) &= L_e(x \to \Theta) \\
&\quad + \int_{\Omega_x} f_r(x, \Psi \to \Theta) L(x \leftarrow \Psi) cos(\mathbf{n}_x, \Psi) d\omega_\Psi \\
&= L_e(x) + \int_{\Omega_x} f_r(x) L(x \leftarrow \Psi) cos(\mathbf{n}_x, \Psi) d\omega_\Psi
\end{aligned}
$$

- Angles does not matter if we have perfectly diffuse surfaces
- Write the rendering equation in terms of surfaces instead

Radiosity

$$
\begin{aligned}
L(x \to \Theta) &= L_e(x \to \Theta) \\
&\quad + \int_{\Omega_x} f_r(x, \Psi \to \Theta) L(x \leftarrow \Psi) cos(\mathbf{n}_x, \Psi) d\omega_\Psi \\
&= L_e(x) + \int_{\Omega_x} f_r(x) L(x \leftarrow \Psi) cos(\mathbf{n}_x, \Psi) d\omega_\Psi \\
&= L_e(x) + \rho(x) \int_S K(x, y) L(y) \mathrm{d}A_y
\end{aligned}
$$

- Angles does not matter if we have perfectly diffuse surfaces
- Write the rendering equation in terms of surfaces instead

Radiosity

$$B_i = \frac{1}{A_i} \int_{S_i} \int_{\Omega_x} L(x \to \Theta) cos(\mathbf{n}_x, \Theta) \mathrm{d}\omega_\Theta \mathrm{d}A_x$$

Radiosity

$$
\begin{aligned}
B_i &= \frac{1}{A_i} \int_{S_i} \int_{\Omega_x} L(x \rightarrow \Theta) cos(\mathbf{n}_x, \Theta) \mathrm{d}\omega_\Theta \mathrm{d}A_x \\
&= \frac{1}{A_i} \int_{S_i} L(x) \int_{\Omega_x} cos(\mathbf{n}_x, \Theta) \mathrm{d}\omega_\Theta \mathrm{d}A_x
\end{aligned}
$$

Radiosity

$$
\begin{aligned}
B_i &= \frac{1}{A_i}\int_{S_i}\int_{\Omega_x} L(x \to \Theta) cos(\mathbf{n}_x, \Theta)\mathrm{d}\omega_\Theta \mathrm{d}A_x \\
&= \frac{1}{A_i}\int_{S_i} L(x)\int_{\Omega_x} cos(\mathbf{n}_x, \Theta)\mathrm{d}\omega_\Theta \mathrm{d}A_x \\
&= \frac{1}{A_i}\int_{S_i} L(x)\pi \mathrm{d}A_x
\end{aligned}
$$

Radiosity

$$
\begin{aligned}
B_i &= \frac{1}{A_i}\int_{S_i}\int_{\Omega_x} L(x \to \Theta) cos(\mathbf{n}_x, \Theta)\mathrm{d}\omega_\Theta \mathrm{d}A_x \\
&= \frac{1}{A_i}\int_{S_i} L(x) \int_{\Omega_x} cos(\mathbf{n}_x, \Theta)\mathrm{d}\omega_\Theta \mathrm{d}A_x \\
&= \frac{1}{A_i}\int_{S_i} L(x)\pi \mathrm{d}A_x \\
&= \frac{1}{A_i}\int_{S_i} B(x) \mathrm{d}A_x
\end{aligned}
$$

Reflectivity

- If we have only perfectly diffuse surfaces we don't need full BRDF

$$\rho(x) \text{ -- reflectivity at point } x$$

- The light that arrives at point x from surface y only depends on visibility

$$K(x, y) \text{ -- how much of light leaving point } y \text{ arrives at } x$$

Radiosity

$$\frac{1}{A_i}\int_{S_i} B(x)\mathrm{d}A_x = \frac{1}{A_i}\int_{S_i} B_e(x)\mathrm{d}A_x + \frac{1}{A_i}\int_{S_i}\rho(x)\int_S K(x,y)B(y)\mathrm{d}A_y\mathrm{d}A_x$$

Assumption the radiosity is constant across a patch

$$\forall x \in S_i : B(x) = B_i$$

Assumption the reflectivity is constant across $\forall x \in S_i : \rho(x) = \rho_i$

Radiosity

$$\frac{1}{A_i}\int_{S_i} B(x)\mathrm{d}A_x = \frac{1}{A_i}\int_{S_i} B_e(x)\mathrm{d}A_x + \frac{1}{A_i}\int_{S_i} \rho(x)\int_S K(x,y)B(y)\mathrm{d}A_y\mathrm{d}A_x$$

$$B_i = B_{ei} + \rho_i \sum_j \frac{1}{A_i}\int_{S_i}\int_{S_j} K(x,y)B(y)\mathrm{d}A_y\mathrm{d}A_x$$

Assumption the radiosity is constant across a patch

$$\forall x \in S_i : B(x) = B_i$$

Assumption the reflectivity is constant across $\forall x \in S_i : \rho(x) = \rho_i$

Radiosity

$$\frac{1}{A_i}\int_{S_i} B(x)\mathrm{d}A_x = \frac{1}{A_i}\int_{S_i} B_e(x)\mathrm{d}A_x + \frac{1}{A_i}\int_{S_i}\rho(x)\int_S K(x,y)B(y)\mathrm{d}A_y\mathrm{d}A_x$$

$$B_i = B_{ei} + \rho_i \sum_j \frac{1}{A_i}\int_{S_i}\int_{S_j} K(x,y)B(y)\mathrm{d}A_y\mathrm{d}A_x$$

$$B_i = \frac{1}{A_i}\int_{S_i} B(x)\mathrm{d}A_x$$

$$\rho_i = \rho(x)$$

Assumption the radiosity is constant across a patch

$$\forall x \in S_i : B(x) = B_i$$

Assumption the reflectivity is constant across $\forall x \in S_i : \rho(x) = \rho_i$

Radiosity

$$B_i = B_{ei} + \rho_i \sum_j \frac{1}{A_i} \int_{S_i} \int_{S_j} K(x,y) B(y) \mathrm{d}A_y \mathrm{d}A_x$$

- Discretise the scene into patches
- $K(x,y)$ constant over patch
- F_{ij} is depends only on the geometry
- referred to as a form factor

Radiosity

$$B_i = B_{ei} + \rho_i \sum_j \frac{1}{A_i} \int_{S_i} \int_{S_j} K(x, y) B(y) \mathrm{d}A_y \mathrm{d}A_x$$
$$= B_{ei} + \rho_i \sum_j F_{ij} B_j$$

- Discretise the scene into patches
- $K(x, y)$ constant over patch
- F_{ij} is depends only on the geometry
- referred to as a form factor

Radiosity

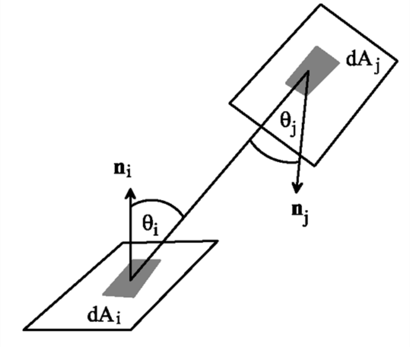
$$B_i = B_{ei} + \rho_i \sum_j \frac{1}{A_i} \int_{S_i} \int_{S_j} K(x, y) B(y) \mathrm{d}A_y \mathrm{d}A_x$$

$$= B_{ei} + \rho_i \sum_j F_{ij} B_j$$

$$F_{ij} = \frac{1}{A_i} \int_{S_i} \int_{S_j} K(x, y) \mathrm{d}A_y \mathrm{d}A_x$$

- Discretise the scene into patches
- $K(x, y)$ constant over patch
- F_{ij} is depends only on the geometry
- referred to as a form factor

Form Factor



- "How large portion of the *view* from *i* is blocked by *j*"

Radiosity

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

$$B_{ei} = B_i - \rho_i \sum_j F_{ij} B_j$$

- The radiosity of patch B_i is a linear combination of radiosities of all other patches B_j
- Importantly the coefficients of this system depends only on geometry which is known and self emitted radiosity which is known

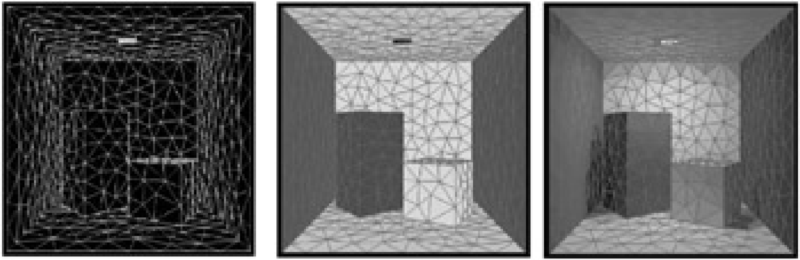
Radiosity

$$B_{ei} = B_i - \rho_i \sum_j F_{ij} B_j$$

$$\begin{bmatrix} B_{e1} \\ B_{e2} \\ \vdots \\ B_{en} \end{bmatrix} = \begin{bmatrix} 1 - \rho_1 F_{11} & -\rho_1 F_{12} & \ldots & \rho_1 F_{1n} \\ -\rho_2 F_{21} & 1 - \rho_2 F_{22} & \ldots & \rho_2 F_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ -\rho_n F_{n1} & -\rho_n F_{n2} & \ldots & 1 - \rho_n F_{nn} \end{bmatrix} \begin{bmatrix} B_1 \\ B_2 \\ \vdots \\ B_n \end{bmatrix}$$

- We know B_{ei} and we can comput F_{ij}
- Solve for B_i

Radiosity



- Compute form factors
- Solve system of equations
- Render image (Raycaster)

Radiosity

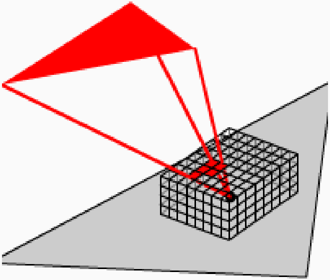
Discretisation Computational and storage cost

Form Factor Computations 1 Cubic storage in number of patches

Form Factor Computations 2 Complicated integral

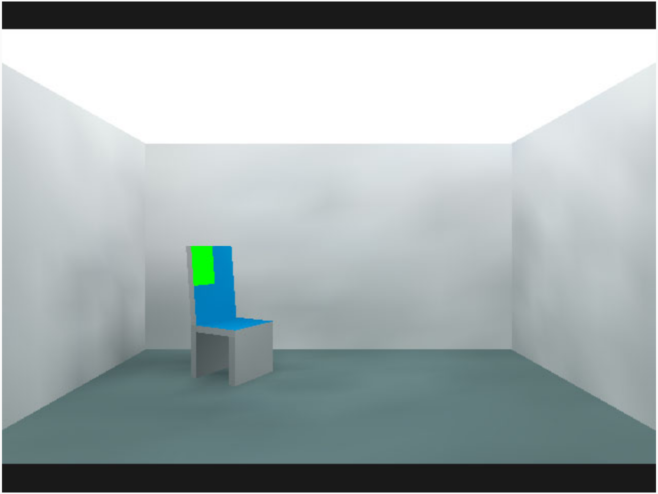
Nummerical Solution Structure in coefficent to exploit

Hemicube approximation



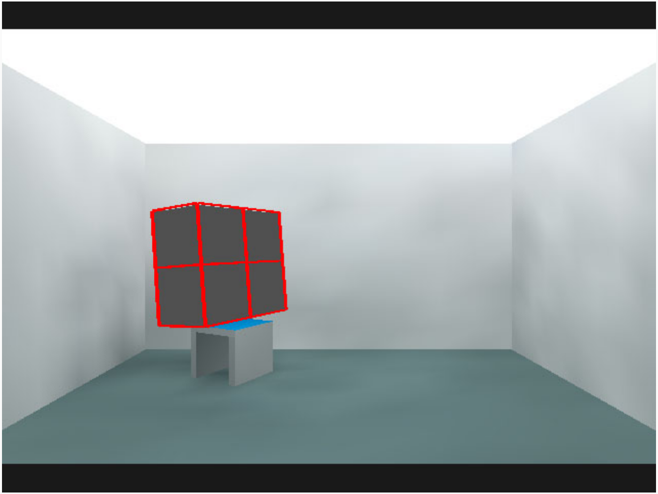
- Form factor computations are expensive
- Approximate with projections onto sphere

Hemicube approximation[3]



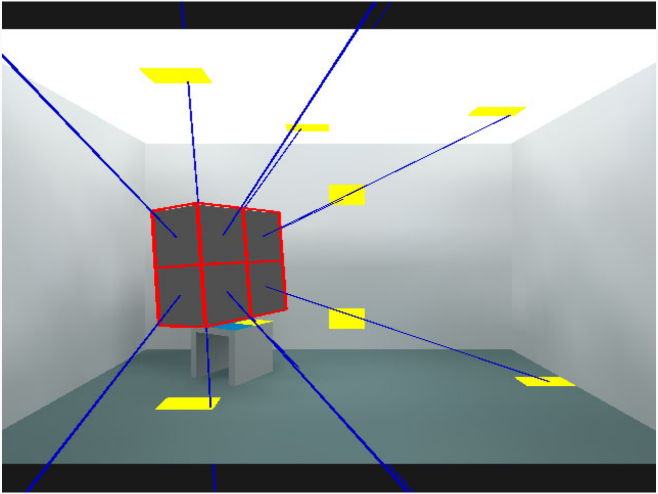
[3] http://www.deluxerender.com/2014/11/hemicube-form-factors/

Hemicube approximation[3]



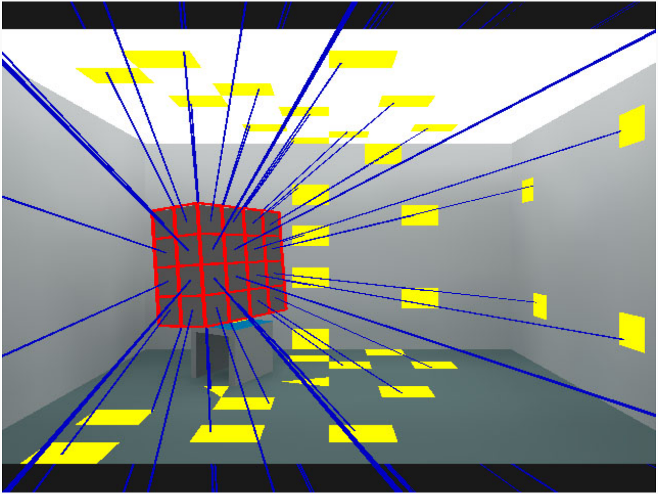
[3] http://www.deluxerender.com/2014/11/hemicube-form-factors/

Hemicube approximation[3]



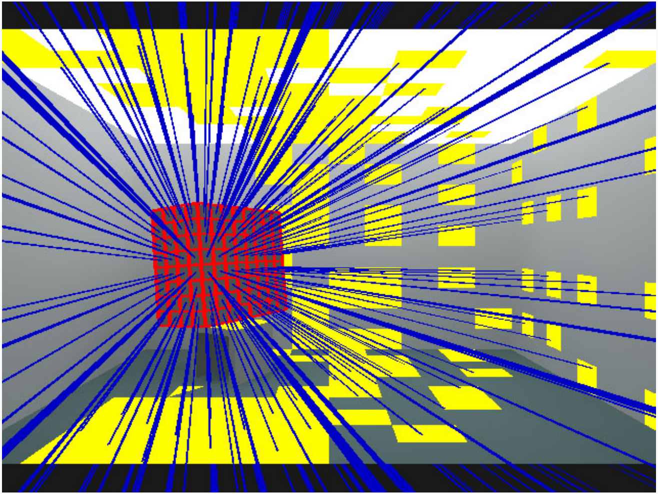
[3] http://www.deluxerender.com/2014/11/hemicube-form-factors/

Hemicube approximation[3]



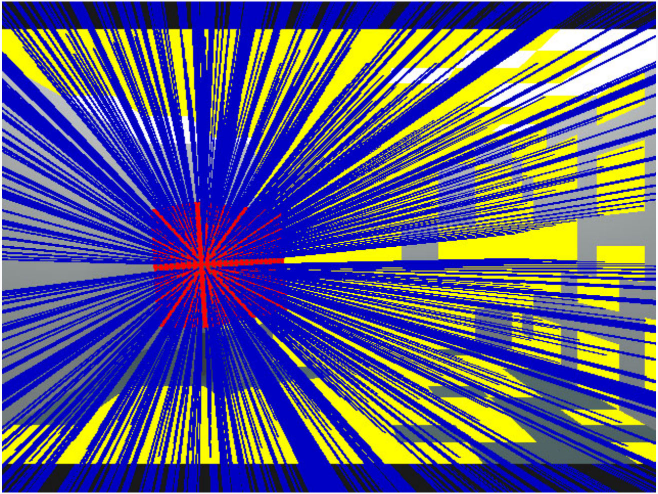
[3] http://www.deluxerender.com/2014/11/hemicube-form-factors/

Hemicube approximation[3]



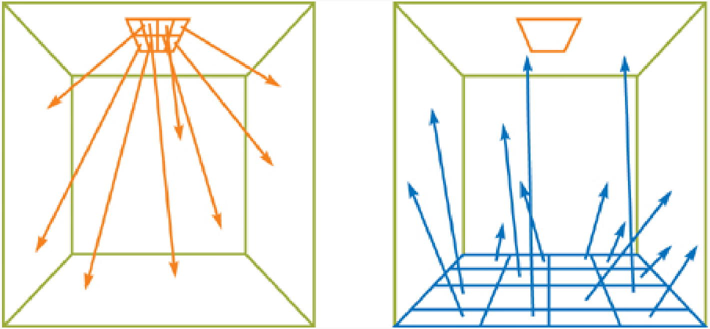
[3] http://www.deluxerender.com/2014/11/hemicube-form-factors/

Hemicube approximation[3]



[3] http://www.deluxerender.com/2014/11/hemicube-form-factors/

Progressive Radiosity[4]



- Solve equation iteratively
- Each patch stores, accumulated and residual energy
- Iterate until residual energy low

[4] http://http.developer.nvidia.com/GPUGems2/gpugems2_chapter39.html

Summary

Summary

- Intuition into rendering equation
- Cast raytracing as a solution to the rendering equation
- Classic Radiosity
    - very clear what assumptions are
    - images look very "radiosity" ;-)

Next Time

Lecture Monday 23rd of March

- Path Tracing (Stochastic Methods)

Lecture Thursday 16th of April

- Photon Mapping (Caustics)

Lecture Monday 27th of April

- Last lecture
- Unit Summary

eof